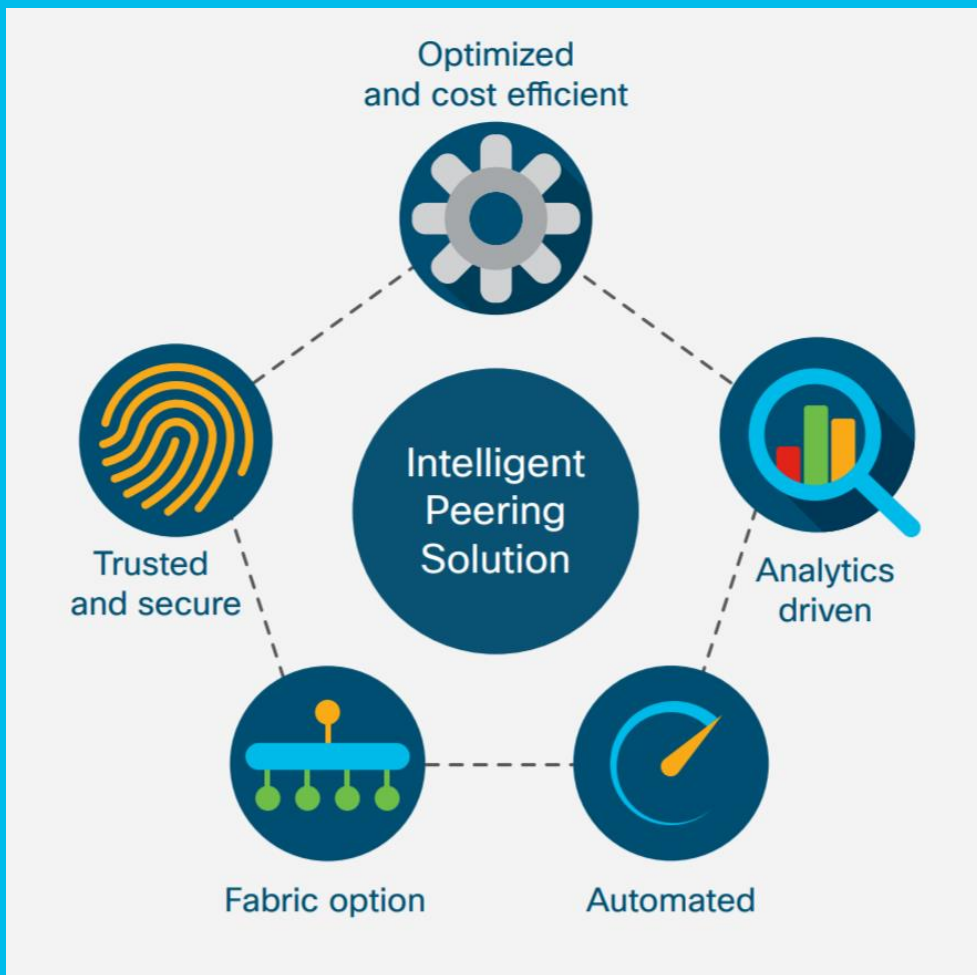




Cisco Intelligent Peering Fabric Solution

hinwoto
technical solutions architect
cisco systems





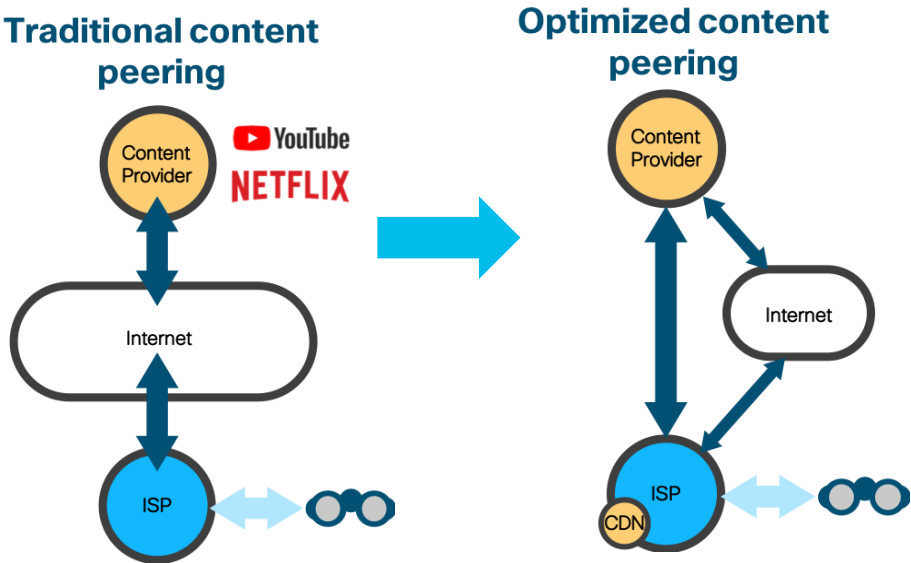
- Peering Trend
- Cisco Peering Solution
 - Benefits & Values
 - Peering Fabric Design
- Demo Peering Fabric Automation
 - Intelligent Peering Traffic Control
 - Automated GW Policy Motion
 - Resiliency with Prioritization

<https://www.cisco.com/c/dam/en/us/solution/s/collateral/service-provider/mobile-internet/solution-overview-c22-741941.pdf>

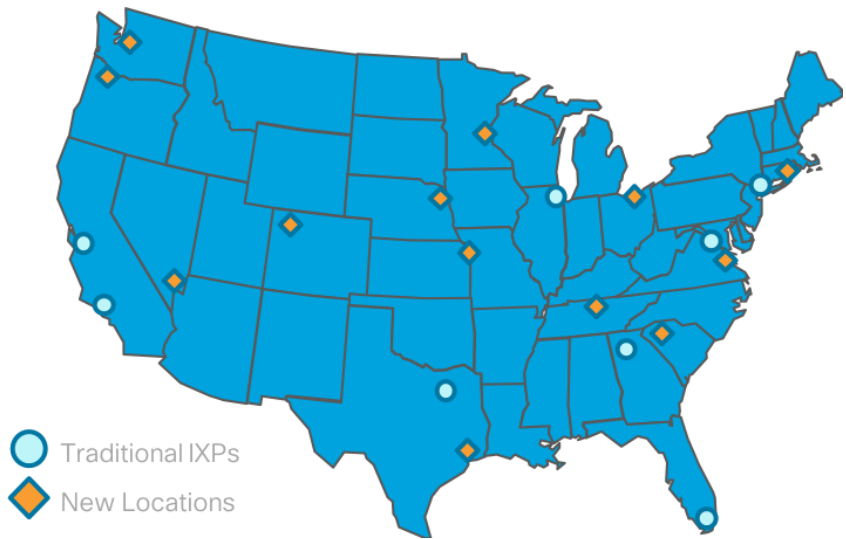
PEERING TREND

DIRECT PEERING & MULTICLOUD

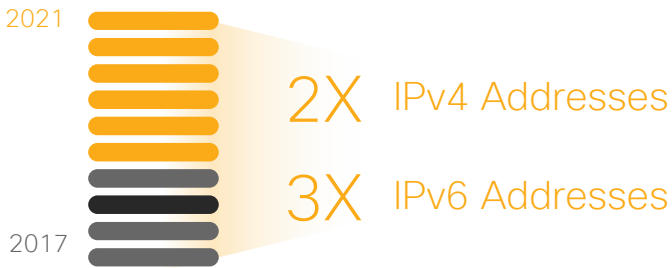
- Traffic moving closer to users (5G, MEC, CDN)
- 82% of Traffic will be video by 2021
- Subscriber BW growing at 31% CAGR
- Stricter peering SLAs with move to SaaS



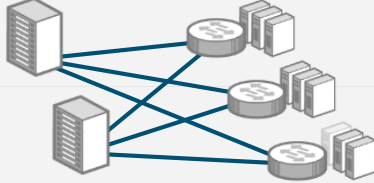
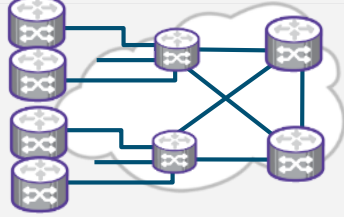
Increased Peering Distribution

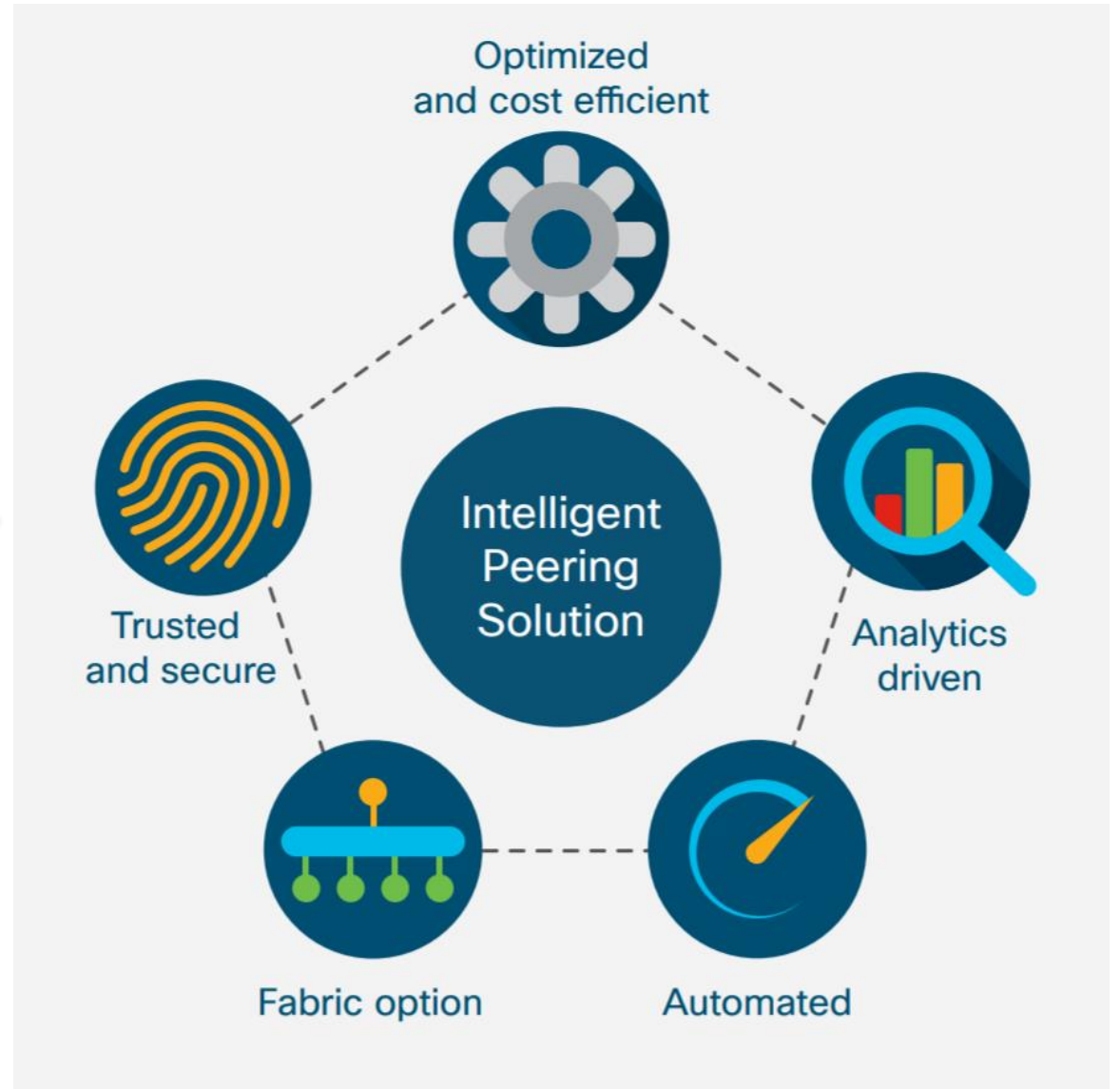
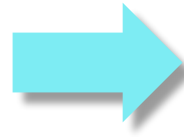


Growing Routing Table



CISCO SERVICE PROVIDER **VALIDATED** DESIGN

End-to-End Designs	Building Blocks
Core	Telemetry and Analytics 
Peering	Segment Routing 
Metro/Access	
Mobile xHaul	
IP and Optical	Scalable Fabric 
Services Data Center	



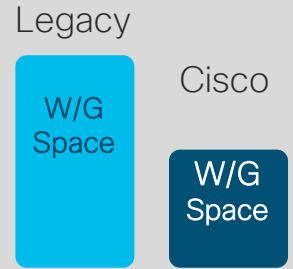
CISCO PEERING FABRIC DESIGN **BENEFITS** OVERVIEW



Validated Design



Faster Time to Revenue
with Automation



Power, Space
& Cooling Savings



Scalable Capacity &
Investment Protection



Telemetry Enabled
Granular Visibility



New level of Security
Focus



Improved Customer
Experience



Fast Convergence
Auto Protection Fabric



CA/OPEX Efficiencies
With Fabric Automation



Higher Reliability
Through
Failure Isolation

CISCO PEERING FABRIC **DESIGN** VALUE PROPOSITION

Cost Optimized

- Dense, price-competitive & **space/power-efficient** Cisco Peering Platform.
- **NSO Automated** configuration & validation.
- Maximize CAPEX **network utilization with Segment Routing**.

Time to Revenue

- **Verified & tested Cisco design** to shorten deployment cycle leveraging best practices.
- Faster **service activation across infrastructure with NSO** & Yang models.

Scalability & Reliability

- Investment protection - **fabric** allowing gradual service expansions **without forklifting**.
- **Fabric isolates failure** domains minimizing traffic impact with single node downtime.
- **Telemetry** unlocks **proactive fault detection** & remediation.

Security

- Dedicated nodes **isolate malicious attacks**.
- Distributed architecture gives more resources for security & monitoring
- **DDoS Protection using BGP Flowspec**

Customer Experience

- **Fast convergence**.
- Cost/power/space – optimized enabling **peering localization closer to customers**.

CISCO PEERING FABRIC DESIGN

Overview

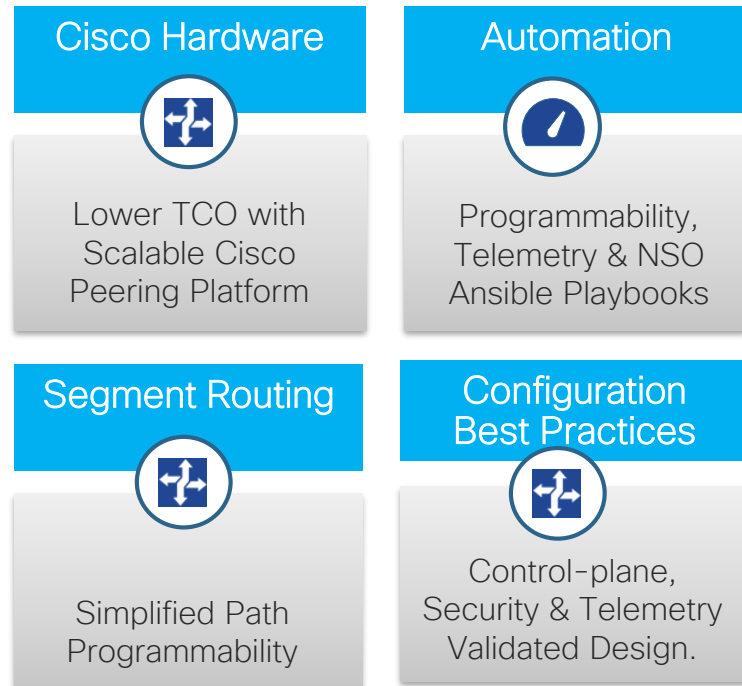
Design Options

Automation

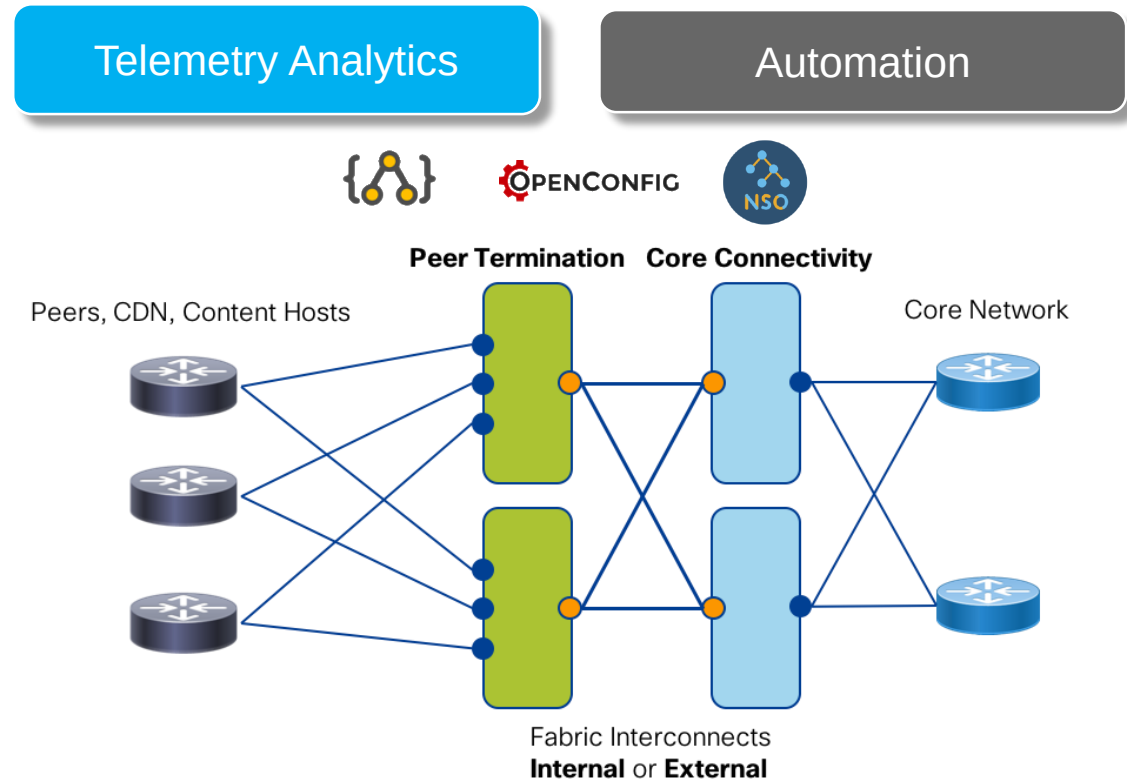
Segment Routing

Best Practice

Design Building Blocks



End-to-End Design



CISCO PEERING FABRIC DESIGN

Overview

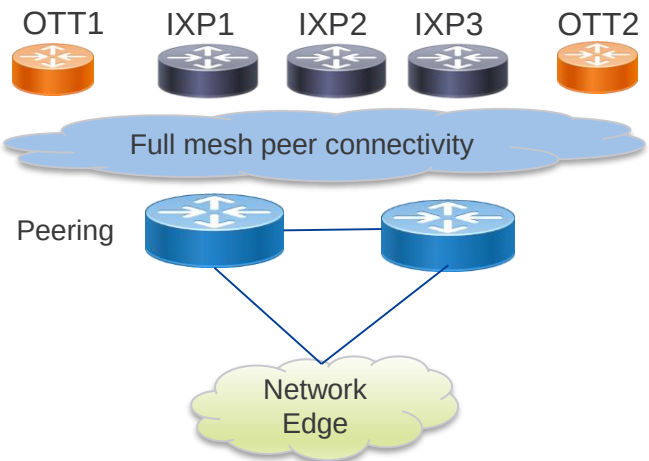
Design Options

Automation

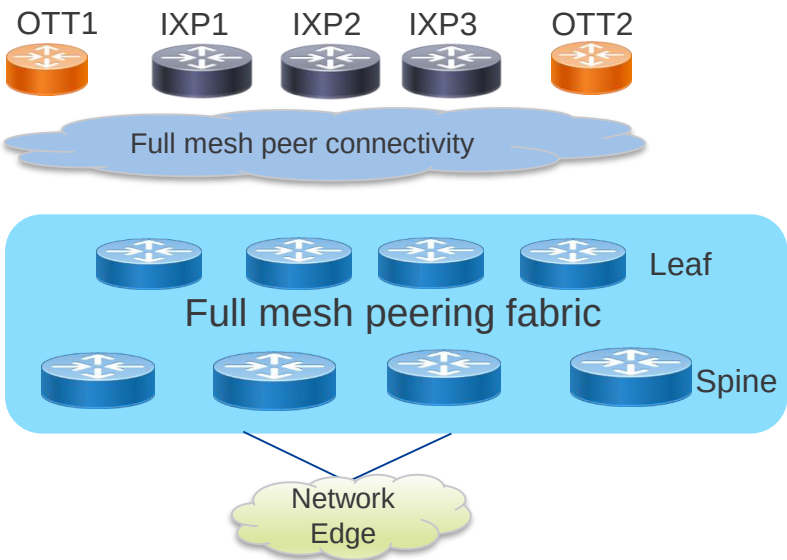
Segment Routing

Best Practice

Option 1: Integrated



Option 2: Distributed



Resiliency

- Single router downtime impacts 50% of traffic

- Router downtime impacts 25% of traffic & decreases as fabric scales.

Peering Visibility

- Security & monitoring resources limited to 2 nodes

- Security & monitoring resources scale with leafs.

Simplify & Prioritize

- Same nodes used for Internet, voice & video.

- Separate peer types (ex. dedicated PE leafs for high priority traffic.)
- Simplified feature set on peering leafs.

Investment Protection

- Future issues with line card compatibility.
- Future higher density might require forklifting.

- Each leaf independent. Future generations seamlessly added to fabric.
- Horizontal scaling saves investments.

CAPEX & OPEX

- Saves costs of optics between leafs and spines.
- Less devices to operate.

- Less critical devices to operate & hence service costs reduced.
- Less critical software upgrades enabling faster innovation.

CISCO PEERING FABRIC DESIGN

Overview

Design Options

Automation

Segment Routing

Best Practice

Automation for Common Tasks

- ❖ NSO Automation
- ❖ OpenConfig YANG models



Network Visualization Analytics
Network Optimization Capacity Planning
Anomaly Detection



Day 0



Day 1



Day 2

- Initial device provisioning

- Control-plane configuration.
- Security ACLs.
- Netflow config & collector.
- Telemetry data & collector.
- BMP receivers.

- New peer turn-up and management.
- Peer capacity expansion.
- PFL fabric node turn-up.

- Intelligent Peering Traffic Control
- Automated GW Policy Motion
- Resiliency with Prioritization

DEMO



CISCO PEERING FABRIC DESIGN

Overview

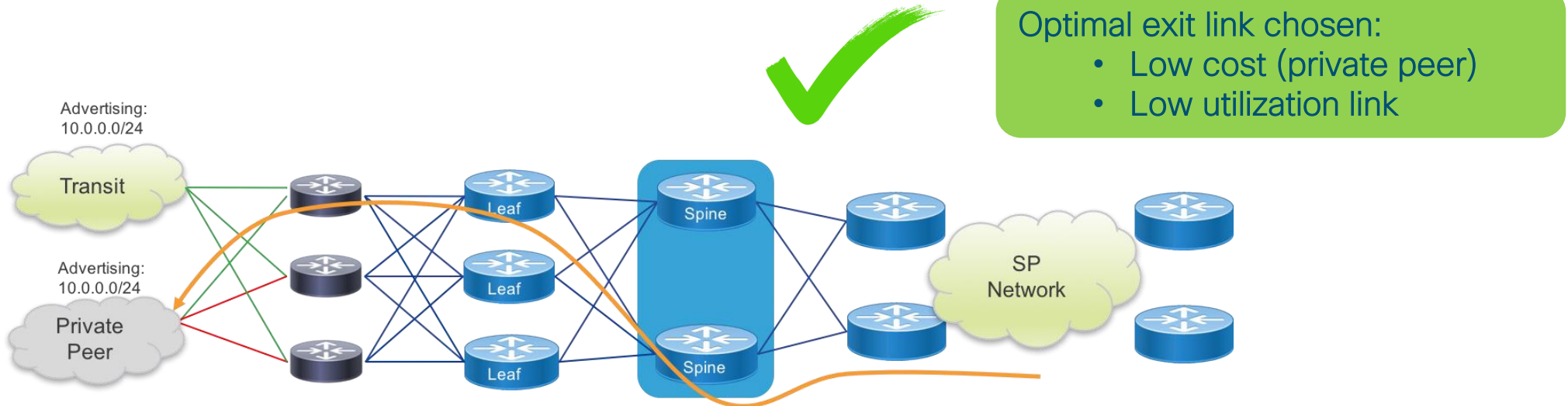
Design Options

Automation

Segment Routing

Best Practice

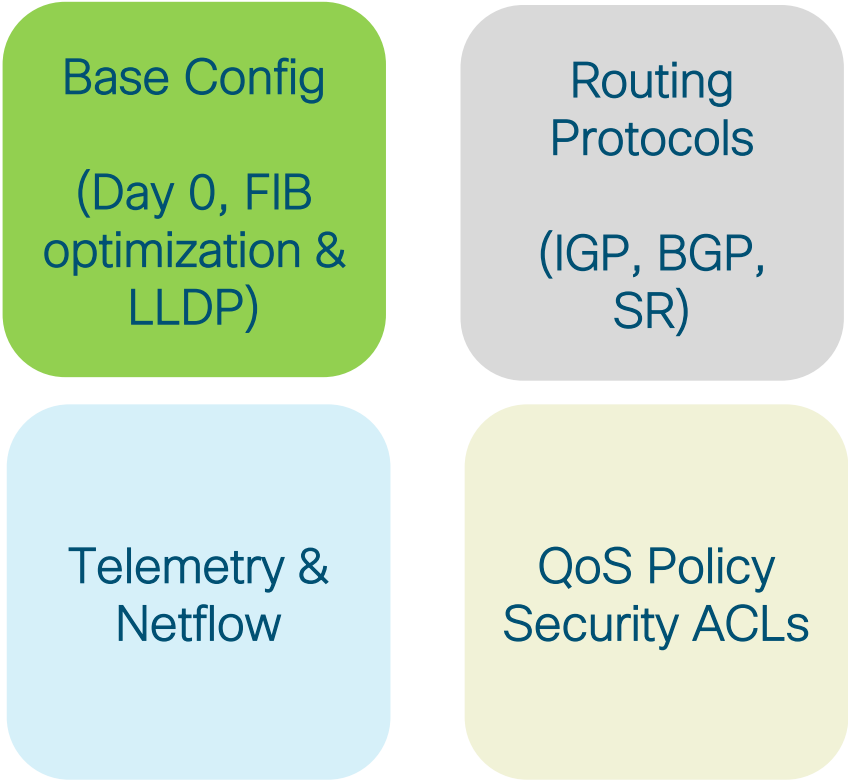
- ❖ Service providers have been facing the following issue in peering networks:
 - **Problem:** Engineering the best network exit path that is cost-efficient while providing good user experience metrics (latency, link utilization & traffic loss).
 - **Solution:** Segment Routing Egress Peer Engineering (EPE).
- ❖ Segment Routing EPE enables Service Providers to dynamically adapt to the best path within their network and the optimal exit, based on the changing network environment.



CISCO PEERING FABRIC DESIGN

❖ Peering configuration best practices that have been tested & validated to ease customer deployments.

Configuration Building Blocks



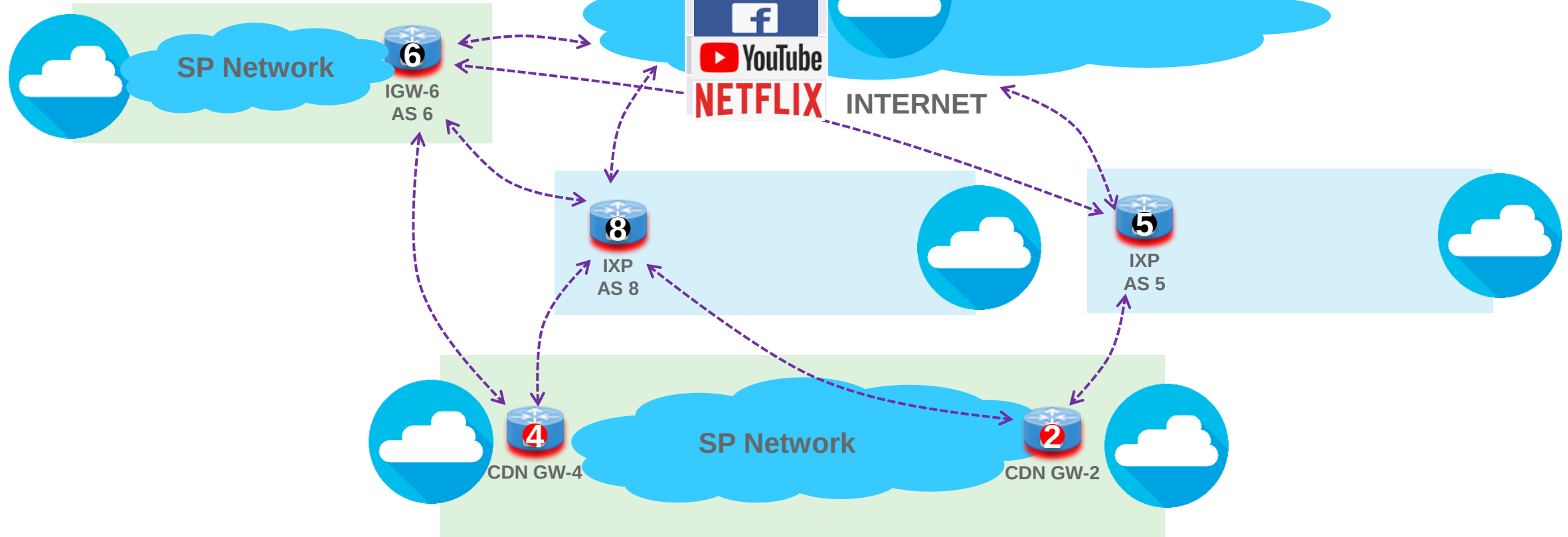
Method	Use Cases
Netflow v9 / IPFIX	• Capacity planning • DDoS detection • Peer traffic anomalies • Enhanced operations
Model-Driven Telemetry	• Capacity planning • Anomaly detection • Automated operational state retrieval
BGP Monitoring Protocol	• BGP prefix anomalies • BGP policy change impact analysis • BGP update statistics • Root-cause analysis for BGP prefix issues



Demo
Peering Fabric Automation:

Intelligent Peering Traffic Control
Automated GW Policy Motion
Resiliency with Prioritization





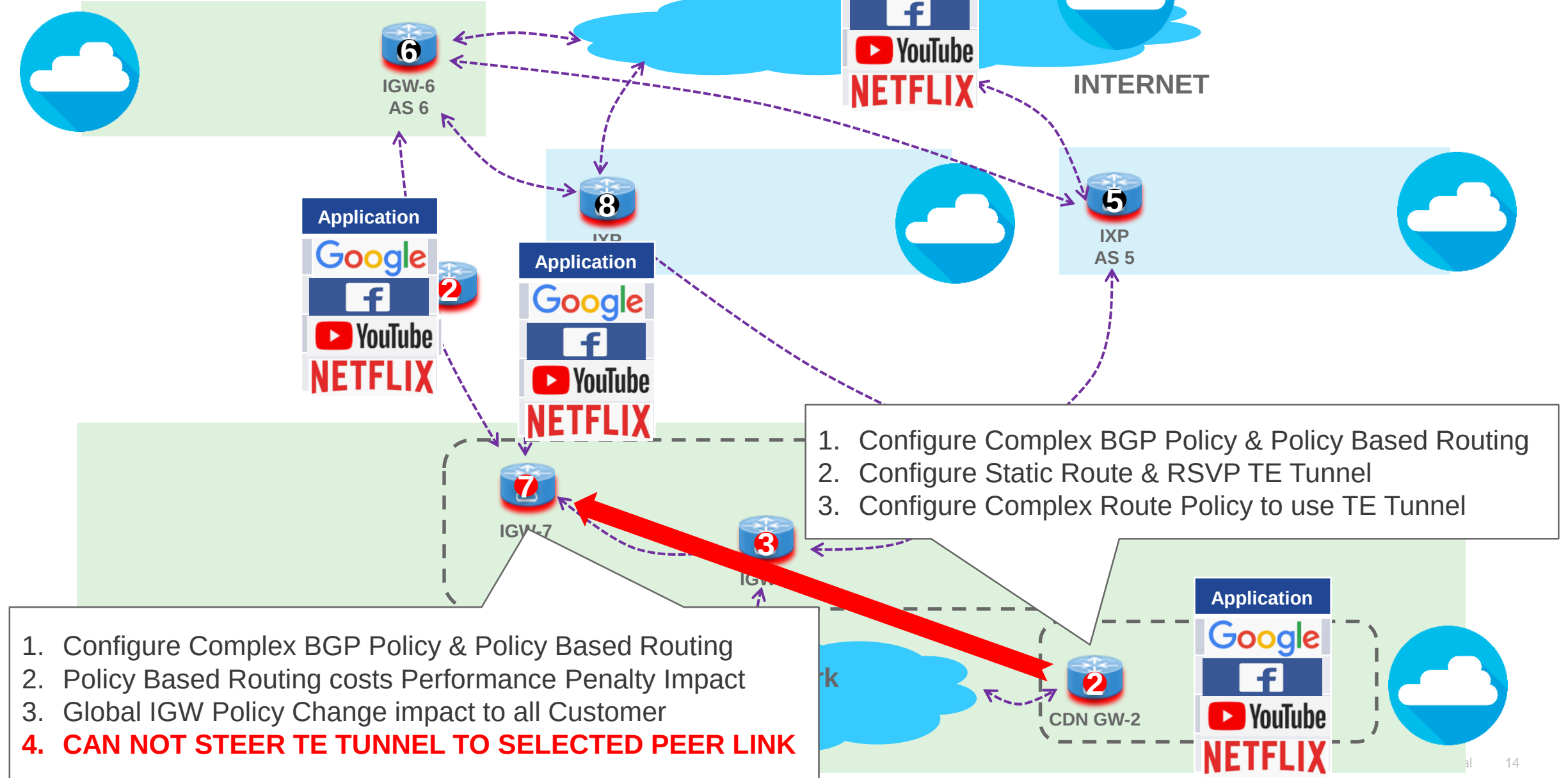
Current Challenge and Complexity : How to steer Specific Traffic to Specific Peer Link

1. Complexity and No Flexibility of Multiple Peer Node and Link Selection
2. Global Customer and Routing Impact for IGW Upstream Policy Change
3. Complex Static Policy Based Routing with Performance Penalty Impact
4. No Resiliency with Prioritization of Multiple Peer Node and Link Selection per Specific Traffic

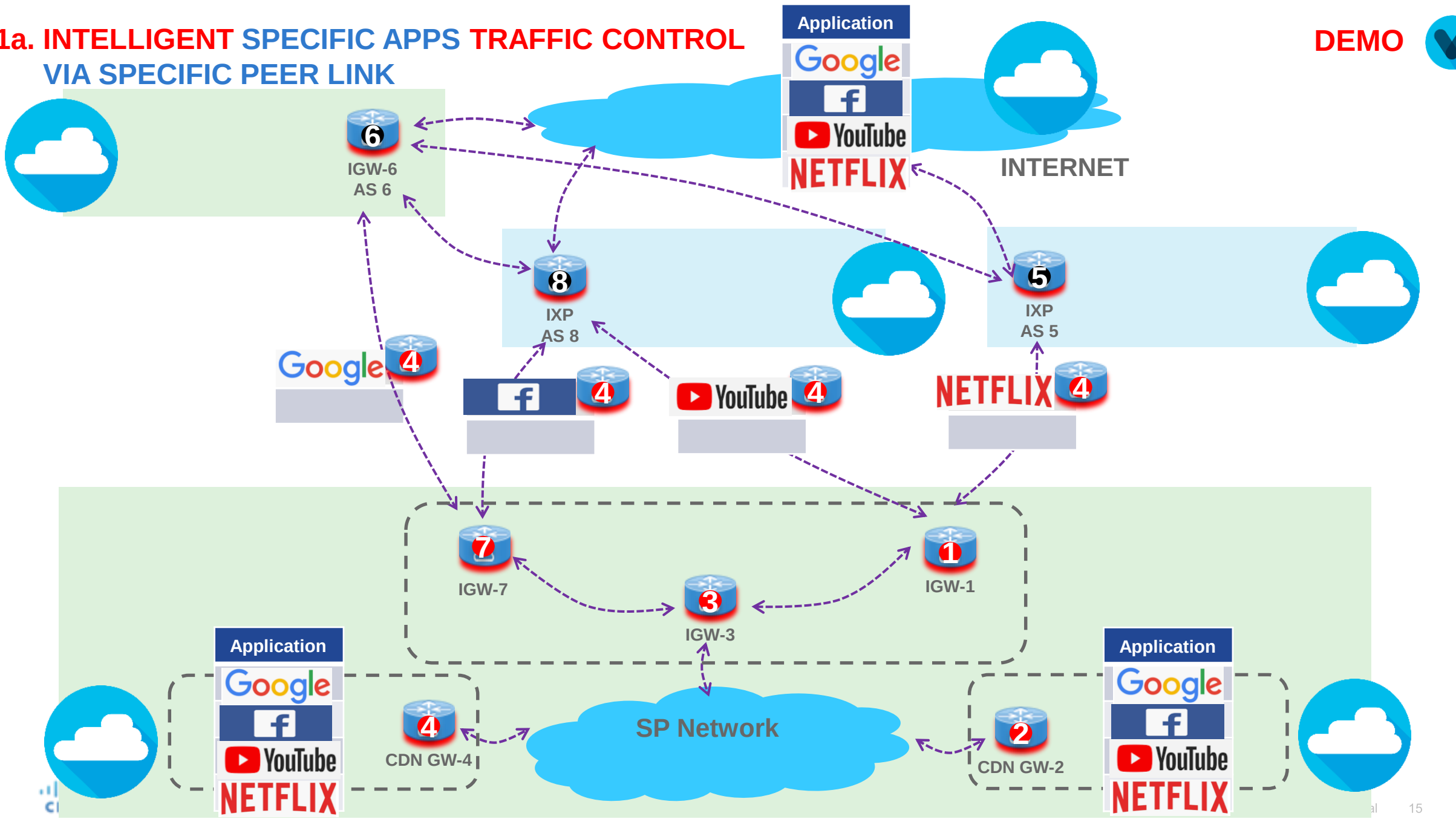
LEGACY WAY TO CONTROL

GOAL : GW2 → GW7 → GW6 → GOOGLE

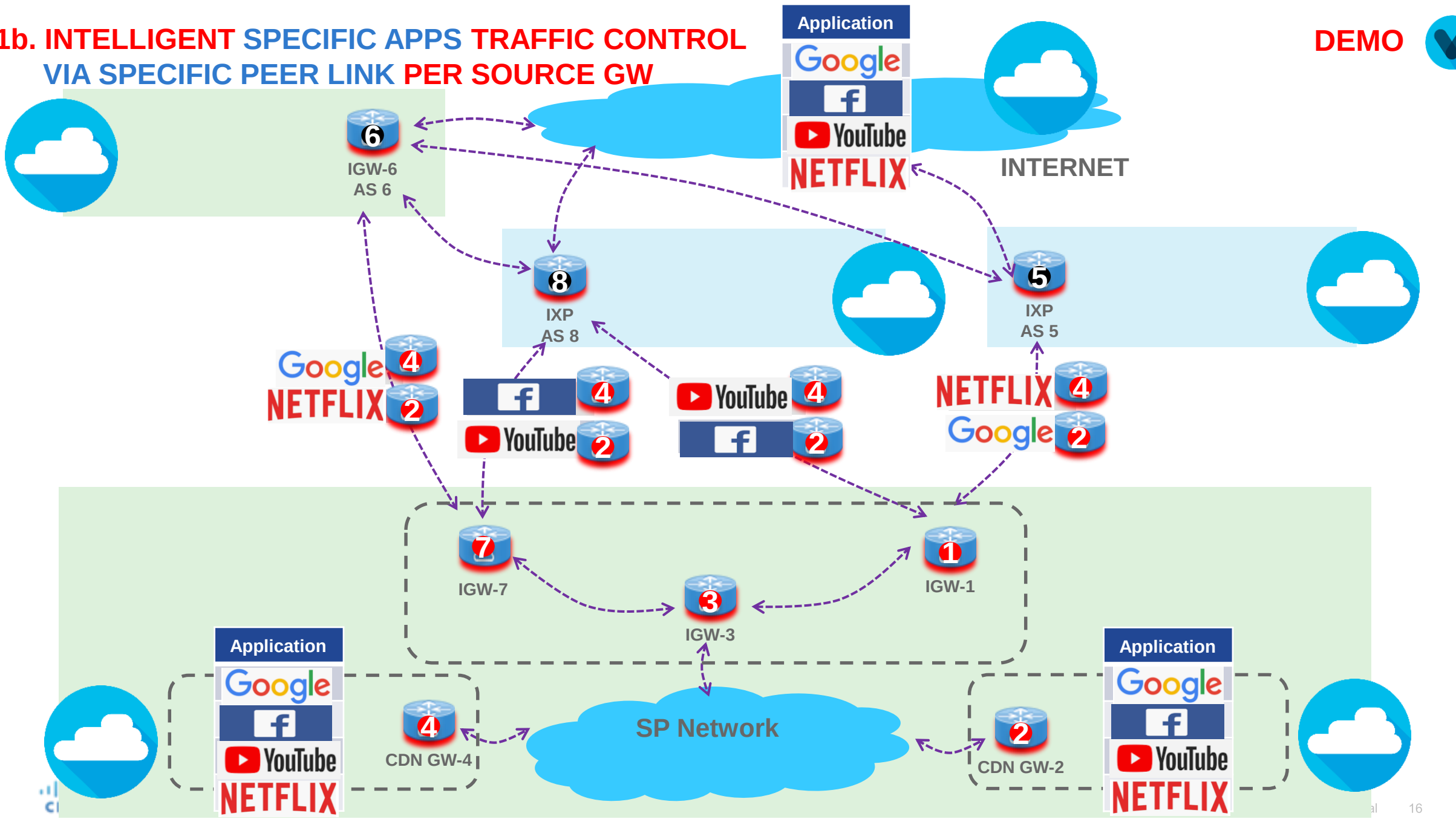
DEMO



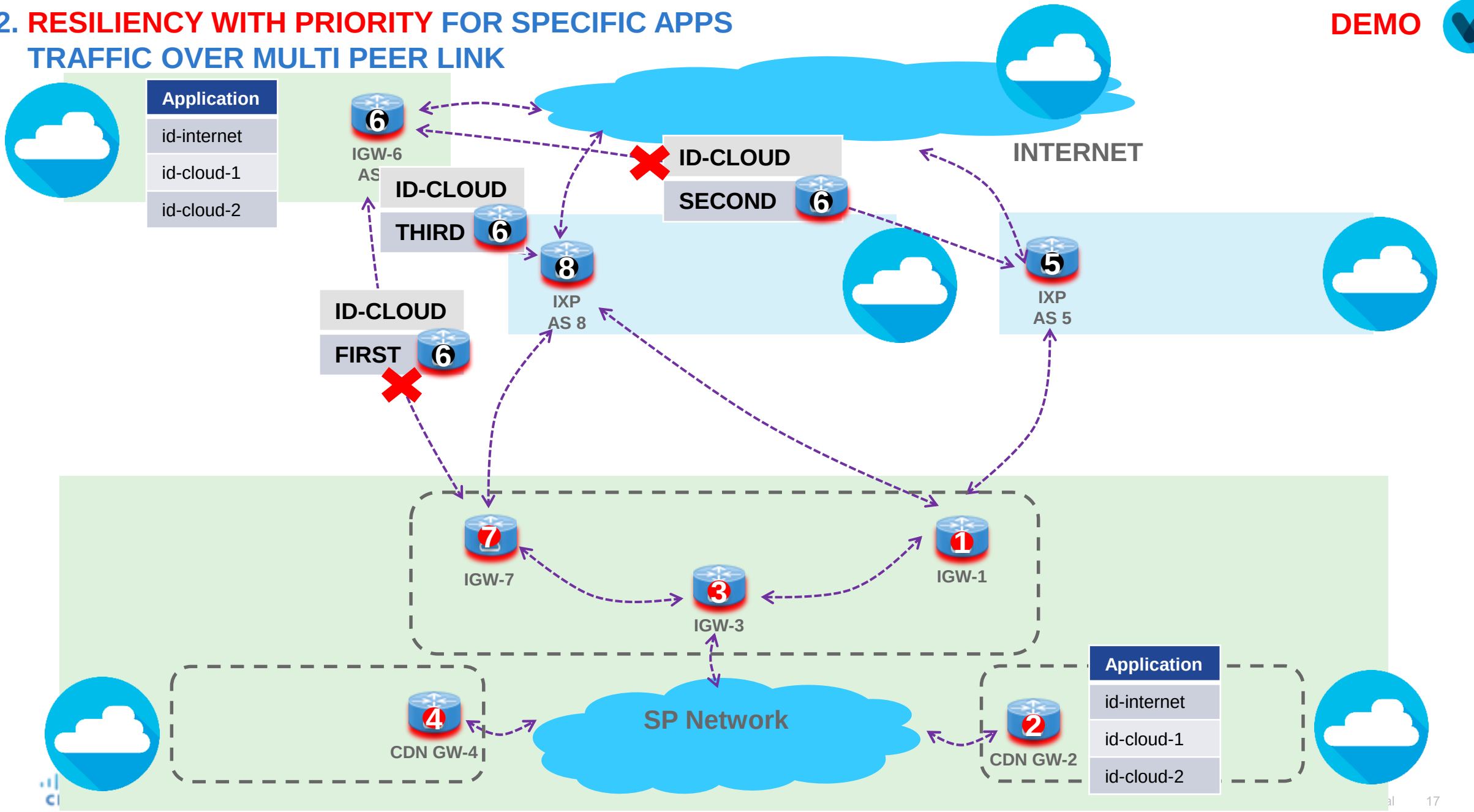
1a. INTELLIGENT SPECIFIC APPS TRAFFIC CONTROL
VIA SPECIFIC PEER LINK



1b. INTELLIGENT SPECIFIC APPS TRAFFIC CONTROL
VIA SPECIFIC PEER LINK PER SOURCE GW



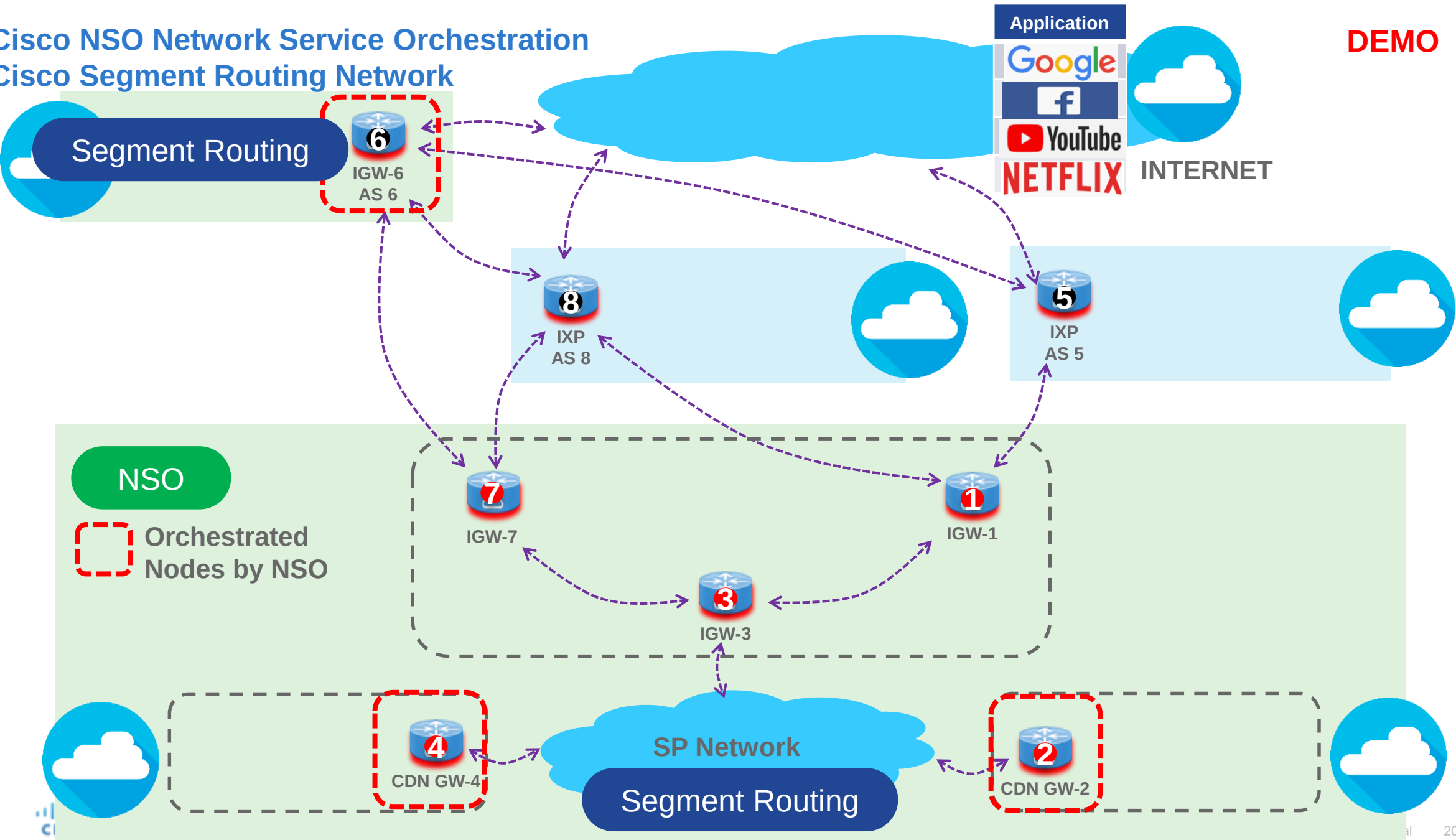
2. RESILIENCY WITH PRIORITY FOR SPECIFIC APPS
TRAFFIC OVER MULTI PEER LINK



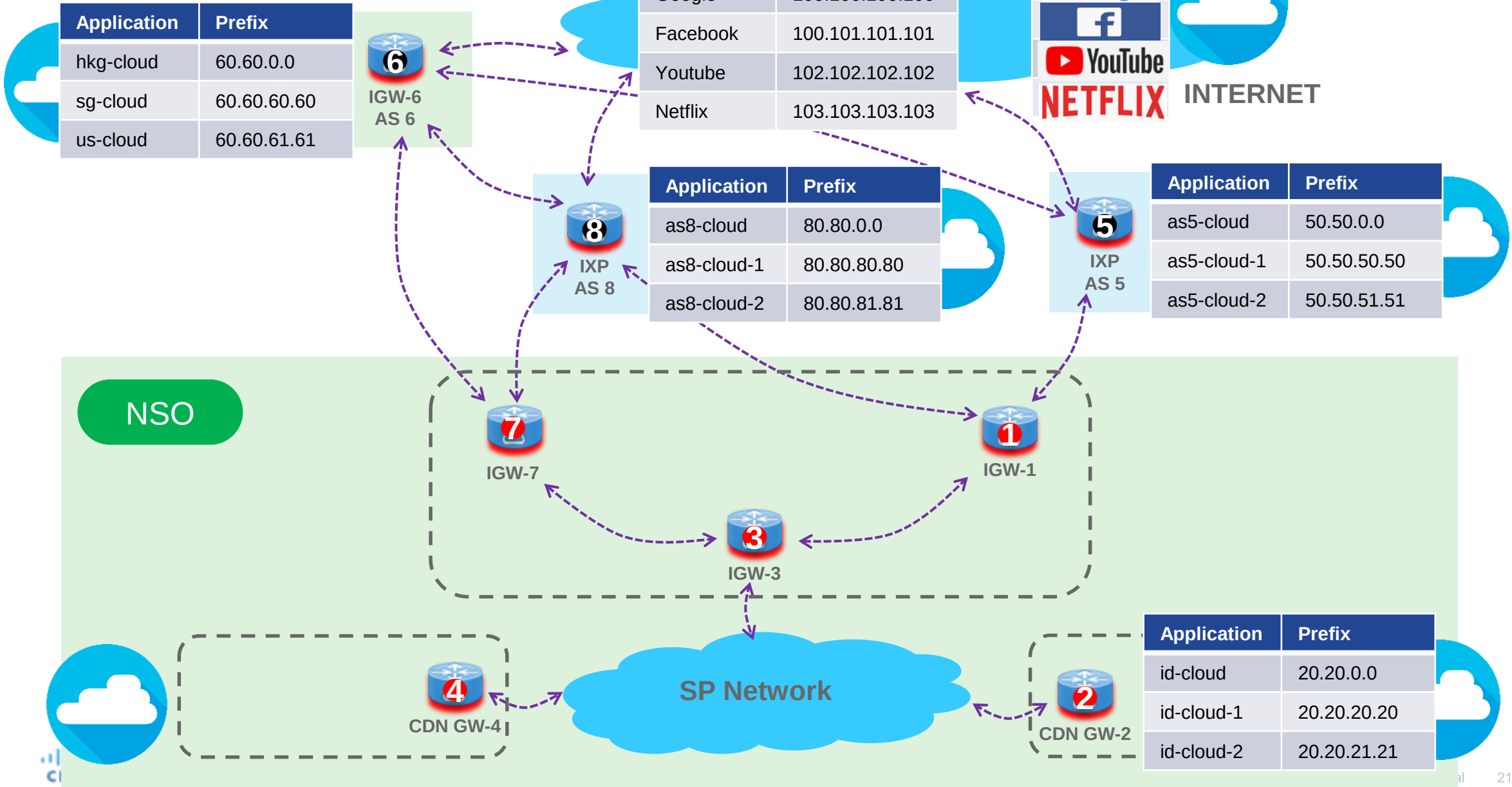


Peering Fabric Network
Cisco Segment Routing Network
Cisco Network Service Orchestration

Cisco NSO Network Service Orchestration
Cisco Segment Routing Network



PEER FABRIC ASN & PREFIX

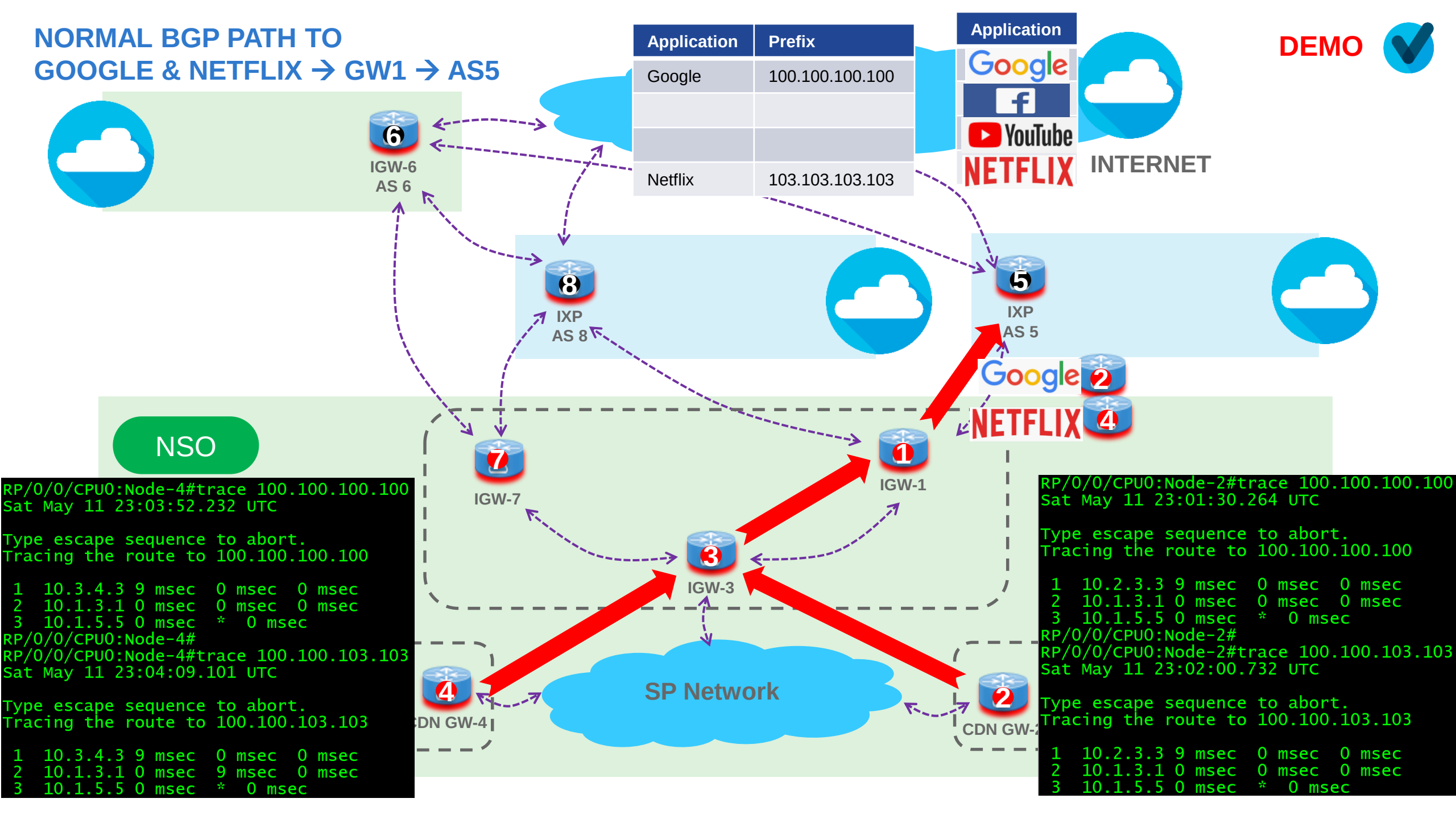




DEMO

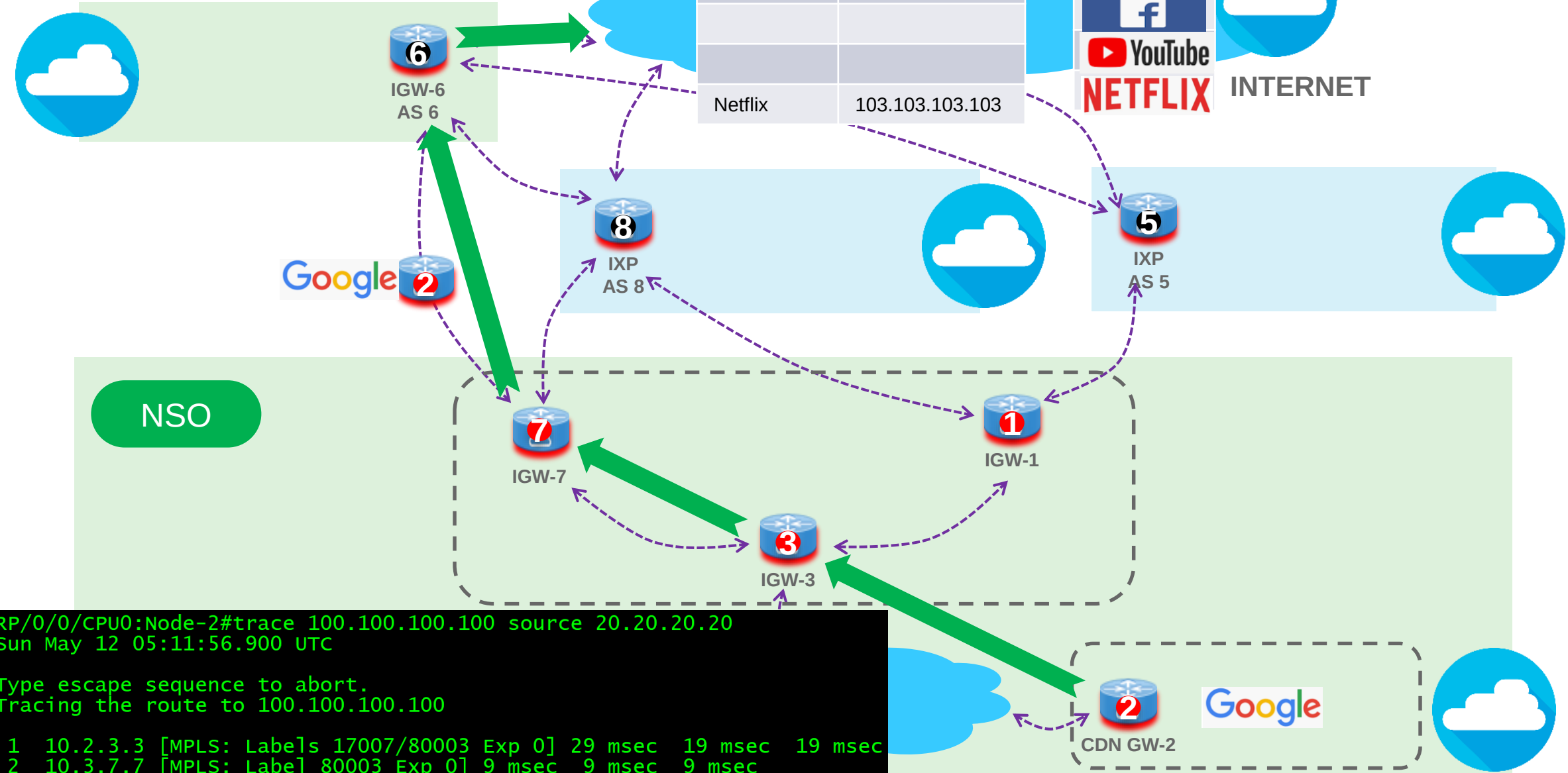
1. Intelligent Traffic Control **Egress** Peering Engineering

NORMAL BGP PATH TO
GOOGLE & NETFLIX → GW1 → AS5



GOOGLE : GW2 → GW7 → AS6 → INTERNET

DEMO 



```
RP/0/0/CPU0:Node-2#trace 100.100.100.100 source 20.20.20.20
Sun May 12 05:11:56.900 UTC

Type escape sequence to abort.
Tracing the route to 100.100.100.100

 1  10.2.3.3 [MPLS: Labels 17007/80003 Exp 0] 29 msec 19 msec 19 msec
 2  10.3.7.7 [MPLS: Label 80003 Exp 0] 9 msec 9 msec 9 msec
 3  10.6.7.6 9 msec 9 msec 9 msec
 4  10.5.6.5 9 msec * 9 msec
```

GOOGLE : GW2 → GW7 → AS6 → INTERNET

Configuration editor /peerir x +

← → ↻ Not secure | 192.168.56.200:8080/webui-one/ConfigurationEditor/peering:peering%7BID-GW-... ☆ 👤 ⋮

 Configuration editor
VERSION: 6.1.1

  View options ▾ root ▾

  /peering:peering{ID-GW-2}/

gw*

node-2

advertised-network

true

path

☐ destination

☐ google_100

```
RP/0/0/CPU0:Node-2#trace www.google.com source id-cld-1
Sun Jun 30 09:13:51.443 UTC

Type escape sequence to abort.
Tracing the route to 100.100.100.100

 1 10.2.3.3 [MPLS: Label 17007/80003 Exp 0] 19 msec  9 msec  9 msec
 2 10.3.7.7 [MPLS: Label 80003 Exp 0] 9 msec  9 msec  9 msec
 3 10.6.7.6 0 msec  9 msec  9 msec
 4 10.5.6.5 9 msec  * 9 msec
RP/0/0/CPU0:Node-2#
RP/0/0/CPU0:Node-2#
RP/0/0/CPU0:Node-2#trace www.netflix.com source id-cld-1
Sun Jun 30 09:14:05.312 UTC

Type escape sequence to abort.
Tracing the route to 103.103.103.103

 1 10.2.3.3 9 msec  19 msec  9 msec
 2 10.1.3.1 0 msec  0 msec  0 msec
 3 10.1.5.5 0 msec  * 9 msec
```

path-1

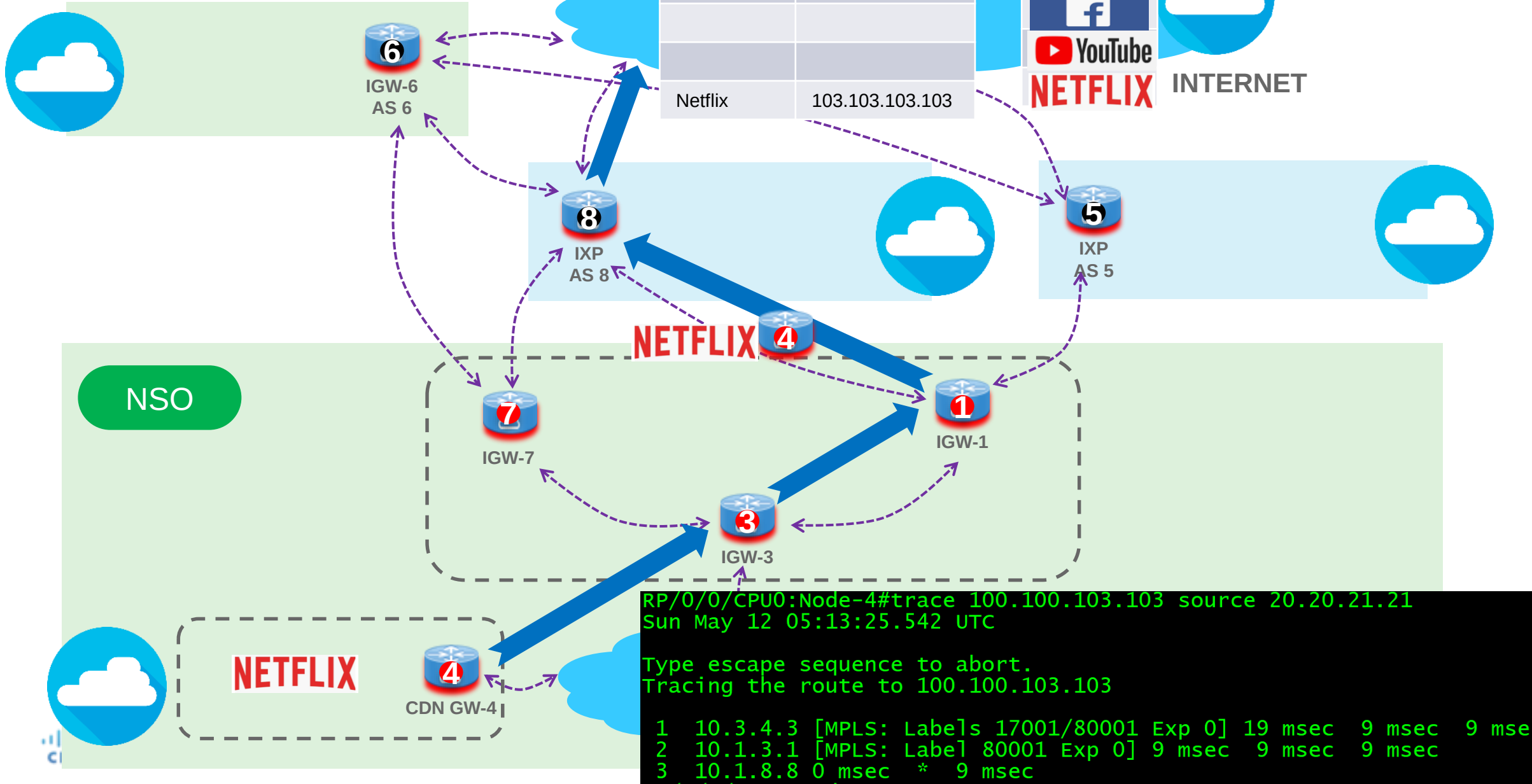
AS6

 Confidential 25

NETFLIX : GW4 → GW1 → AS8 → INTERNET

DEMO 



```
RP/0/0/CPU0:Node-4#trace 100.100.103.103 source 20.20.21.21
Sun May 12 05:13:25.542 UTC

Type escape sequence to abort.
Tracing the route to 100.100.103.103

 1  10.3.4.3 [MPLS: Label 17001/80001 Exp 0] 19 msec  9 msec  9 msec
 2  10.1.3.1 [MPLS: Label 80001 Exp 0] 9 msec  9 msec  9 msec
 3  10.1.8.8 0 msec  *  9 msec
```

NETFLIX : GW4 → GW1 → AS8 → INTERNET

Configuration editor /peerir x

Not secure | 192.168.56.200:8080/webui-one/ConfigurationEditor/peering:peering%7BID-GW-...

CISCO Configuration editor VERSION: 5.1.1

View options root

/peering:peering(ID-GW-4)/

gw*

node-4

advertised-network

true

path

destination

path-1

netflix_103

AS8-1

RP/0/0/CPU0:Node-4#trace www.google.com source id-cloud-2
Sun Jun 30 09:12:43.606 UTC

Type escape sequence to abort.
Tracing the route to 100.100.100.100

 1 10.3.4.3 0 msec 0 msec 0 msec
 2 10.1.3.1 0 msec 0 msec 0 msec
 3 10.1.5.5 19 msec * 0 msec
RP/0/0/CPU0:Node-4#
RP/0/0/CPU0:Node-4#
RP/0/0/CPU0:Node-4#trace www.netflix.com source id-cloud-2
Sun Jun 30 09:12:49.356 UTC

Type escape sequence to abort.
Tracing the route to 103.103.103.103

 1 10.3.4.3 [MPLS: Label 17001/80003 Exp 0] 9 msec 9 msec 0 msec
 2 10.1.3.1 [MPLS: Label 80003 Exp 0] 9 msec 9 msec 9 msec
 3 10.1.8.8 9 msec * 9 msec

27

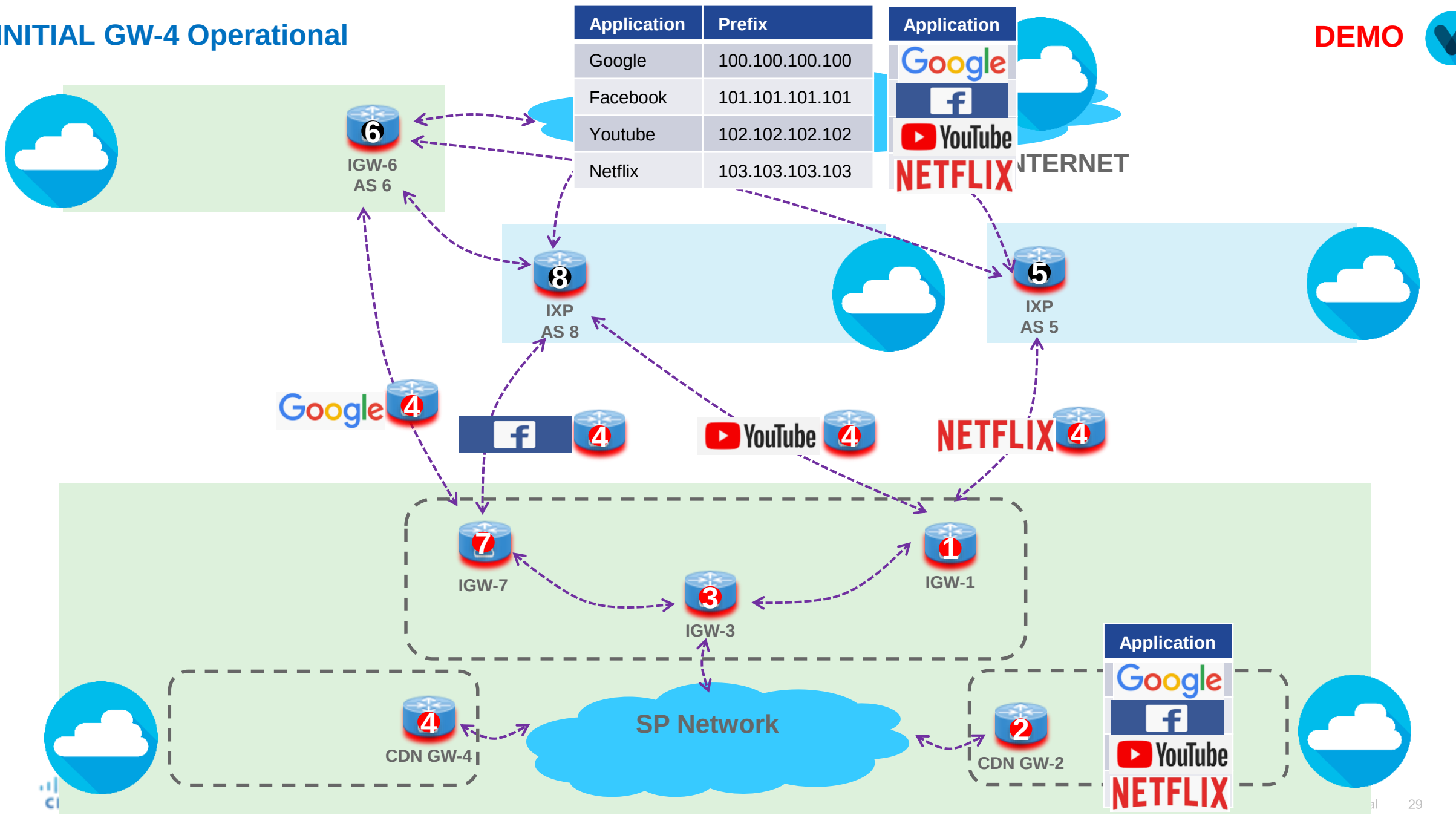
Confidential



DEMO

2. Automated GW Policy Motion

INITIAL GW-4 Operational



INITIAL GW-4 Operational



Configuration editor

VERSION: 5.1.1

[/peering:peering\(ID-GW\)/domestic/path/](#)

path

☐	destination	path-1
☐	google_100	AS6
☒	facebook_101	AS8-7
☐	youtube_102	AS8-1
☐	netflix_103	AS5



```
RP/0/0/CPU0:Node-4#trace www.google.com source id-cloud-1
Sun Jun 30 10:02:35.260 UTC
```

```
Type escape sequence to abort.
Tracing the route to 100.100.100.100
```

```
 1  10.3.4.3 [MPLS: Labels 17007/80003 Exp 0] 9 msec  9 msec  9 msec
 2  10.3.7.7 [MPLS: Label 80003 Exp 0] 9 msec  9 msec  9 msec
 3  10.6.7.6 9 msec  9 msec  9 msec
 4  10.5.6.5 9 msec  *  9 msec
```

```
RP/0/0/CPU0:Node-4#
```

```
RP/0/0/CPU0:Node-4#trace www.facebook.com source id-cloud-1
Sun Jun 30 10:02:52.069 UTC
```

```
Type escape sequence to abort.
Tracing the route to 101.101.101.101
```

```
 1  10.3.4.3 [MPLS: Labels 17007/80143 Exp 0] 19 msec  9 msec  9 msec
 2  10.3.7.7 [MPLS: Label 80143 Exp 0] 9 msec  9 msec  9 msec
 3  10.7.8.8 9 msec  *  9 msec
```

```
RP/0/0/CPU0:Node-4#trace www.youtube.com source id-cloud-1
Sun Jun 30 10:04:34.032 UTC
```

```
Type escape sequence to abort.
Tracing the route to 102.102.102.102
```

```
 1  10.3.4.3 [MPLS: Labels 17001/80003 Exp 0] 9 msec  9 msec  9 msec
 2  10.1.3.1 [MPLS: Label 80003 Exp 0] 9 msec  9 msec  9 msec
 3  10.1.8.8 9 msec  *  9 msec
```

```
RP/0/0/CPU0:Node-4#
```

```
RP/0/0/CPU0:Node-4#trace www.netflix.com source id-cloud-1
Sun Jun 30 10:04:48.631 UTC
```

```
Type escape sequence to abort.
Tracing the route to 103.103.103.103
```

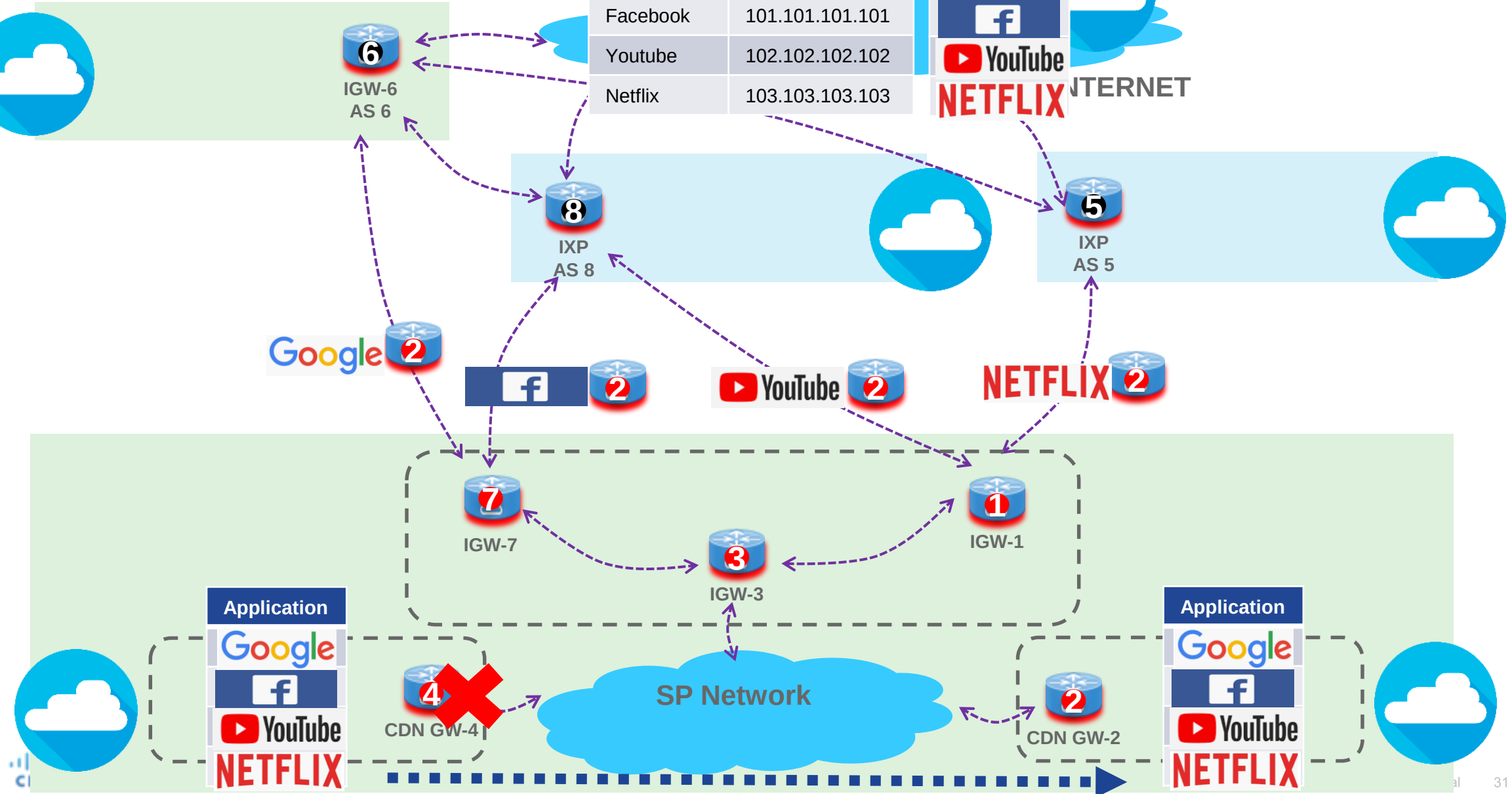
```
 1  10.3.4.3 [MPLS: Labels 17001/80141 Exp 0] 9 msec  9 msec  9 msec
 2  10.1.3.1 [MPLS: Label 80141 Exp 0] 9 msec  9 msec  9 msec
 3  10.1.5.5 9 msec  *  9 msec
```

Automated GW Policy Motion
From Failure GW-2 to GW-4

Application	Prefix
Google	100.100.100.100
Facebook	101.101.101.101
Youtube	102.102.102.102
Netflix	103.103.103.103

Application
Google
Facebook
YouTube
NETFLIX

INTERNET





Configuration editor

VERSION: 5.1.1

[Home](#) [Up](#) /peering:peering{ID-GW}/

ID-GW

gw-site*

domestic-gw

domestic/

gw*

node-4

node-2

node-4

advertised-network

true

C*

Commit manager

E

Configuration editor

B

Dashboard



Configuration editor

VERSION: 5.1.1

[Home](#) [Up](#) /peering:peering{ID-GW}/

name

ID-GW

gw-site*

domestic-gw

domestic/

gw*

node-2

C

Commit manager

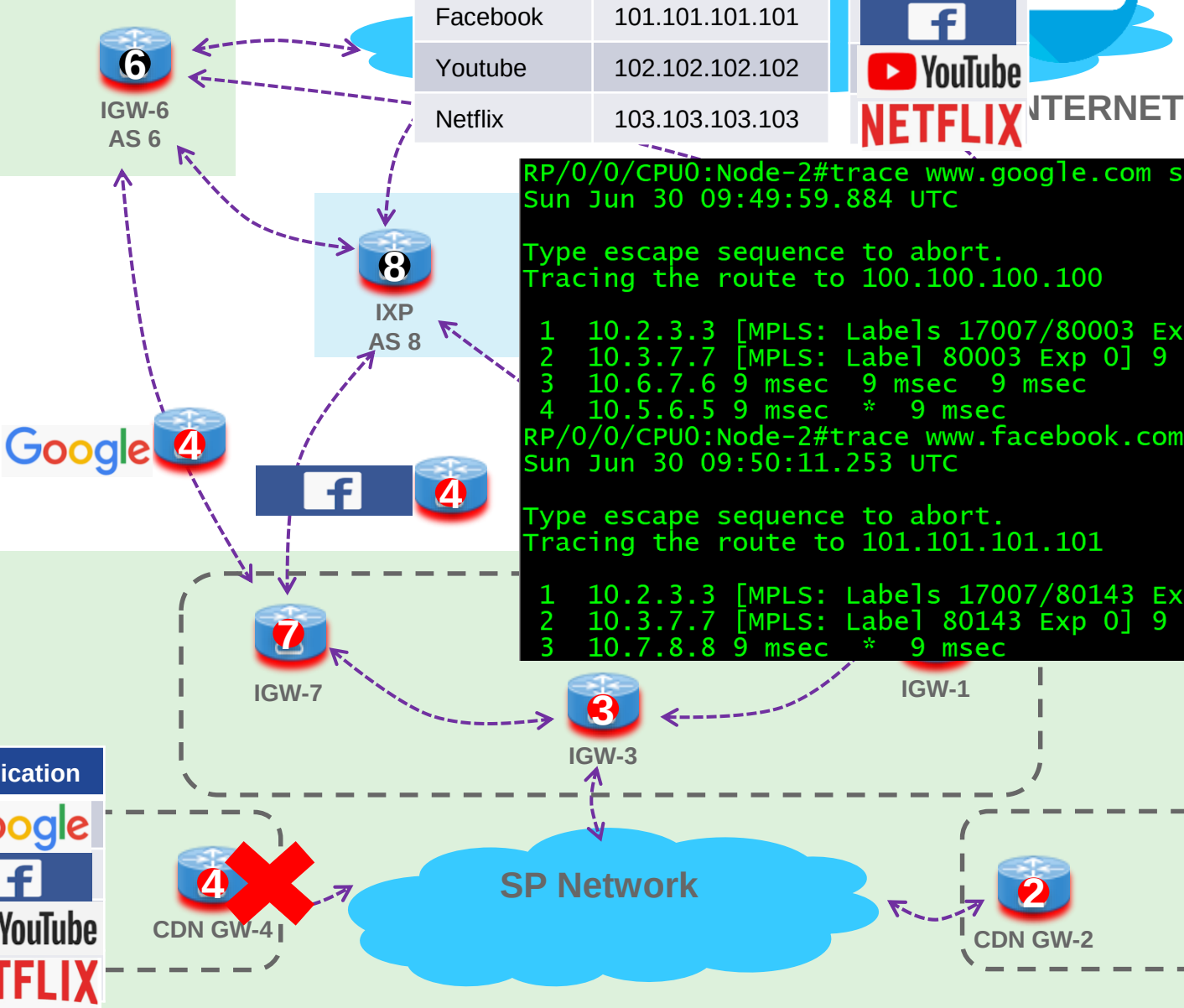
E

Configuration editor

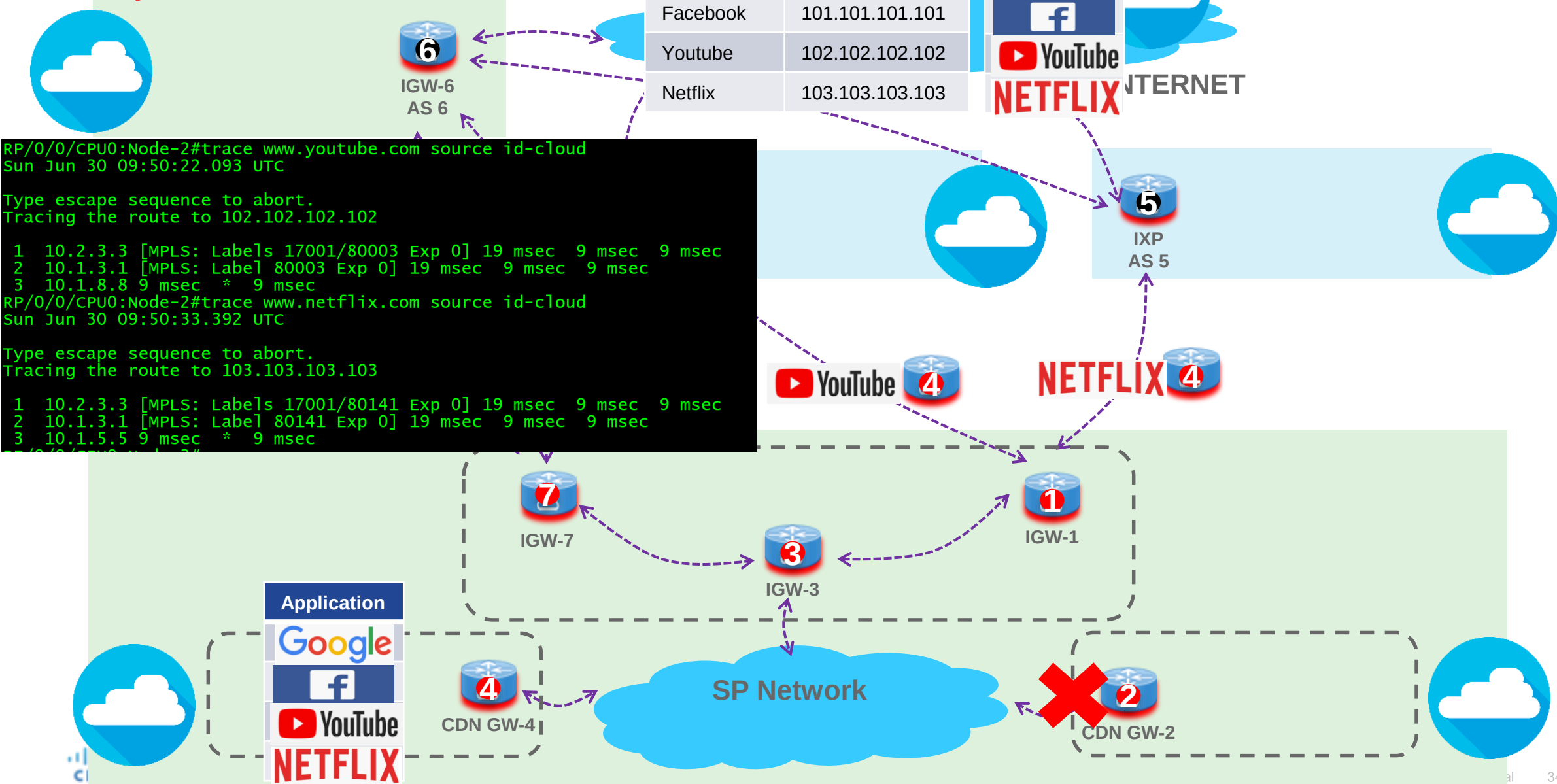
B

Dashboard

Automated GW Policy Motion
GW-2 Operational



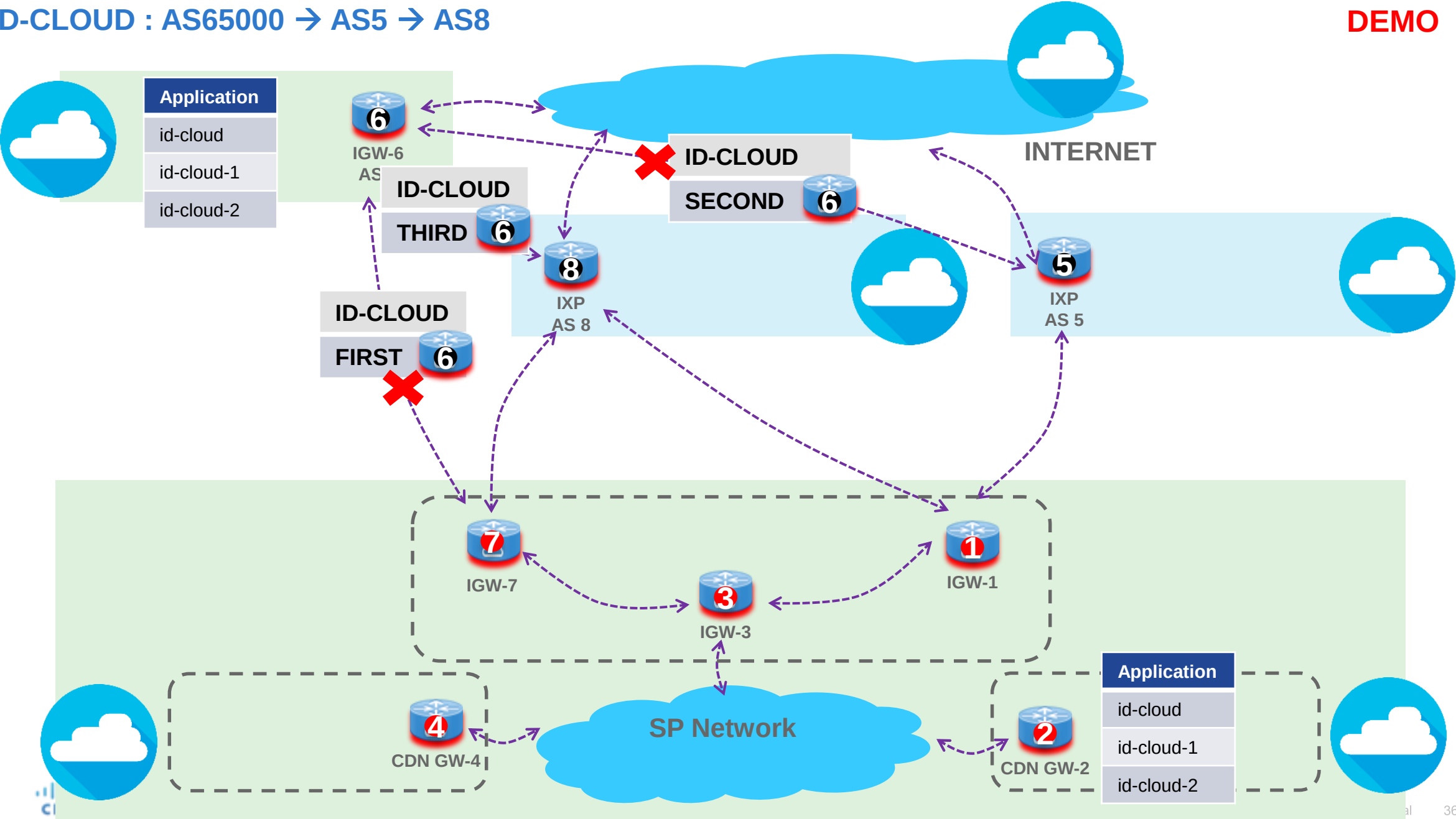
Automated GW Policy Motion
GW-2 Operational





DEMO

3. Resiliency with Prioritization





🏠 ↗ /peering:peering{SG-GW-6}/

gw*



node-6 ▼

advertised-network



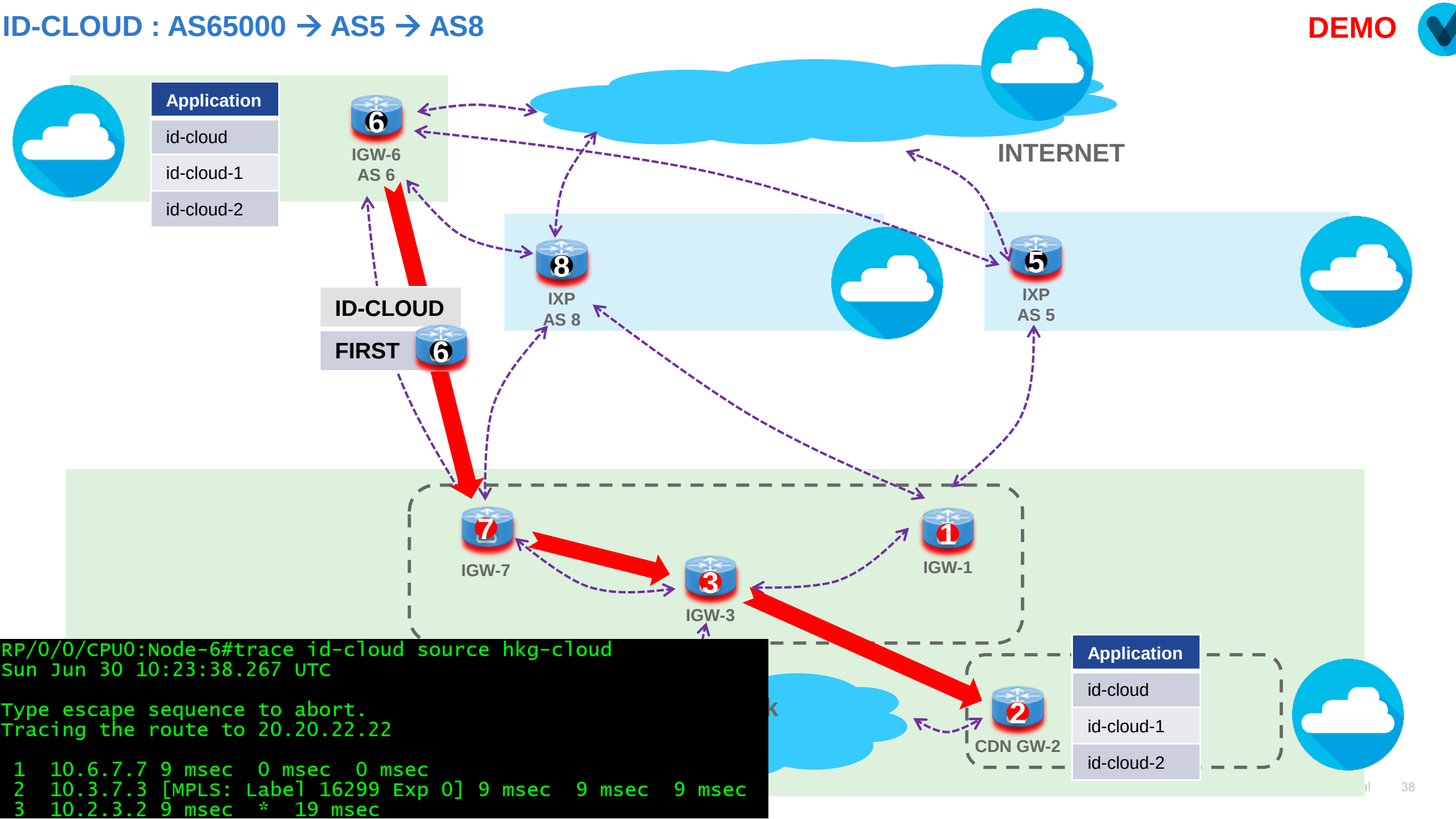
true ▼

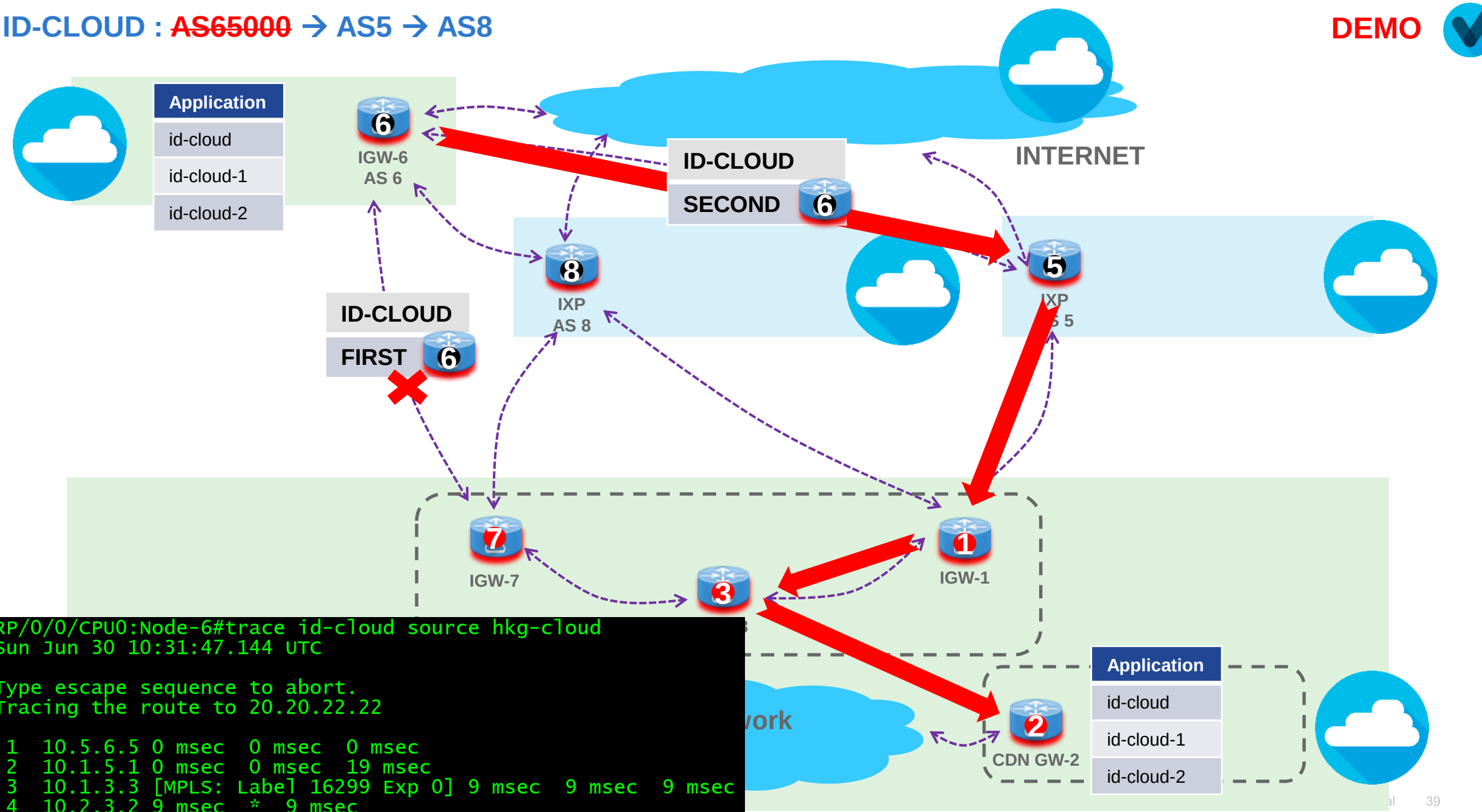
path



☐	destination	path-1	path-2
☐	id-cloud_200	AS65000	AS5



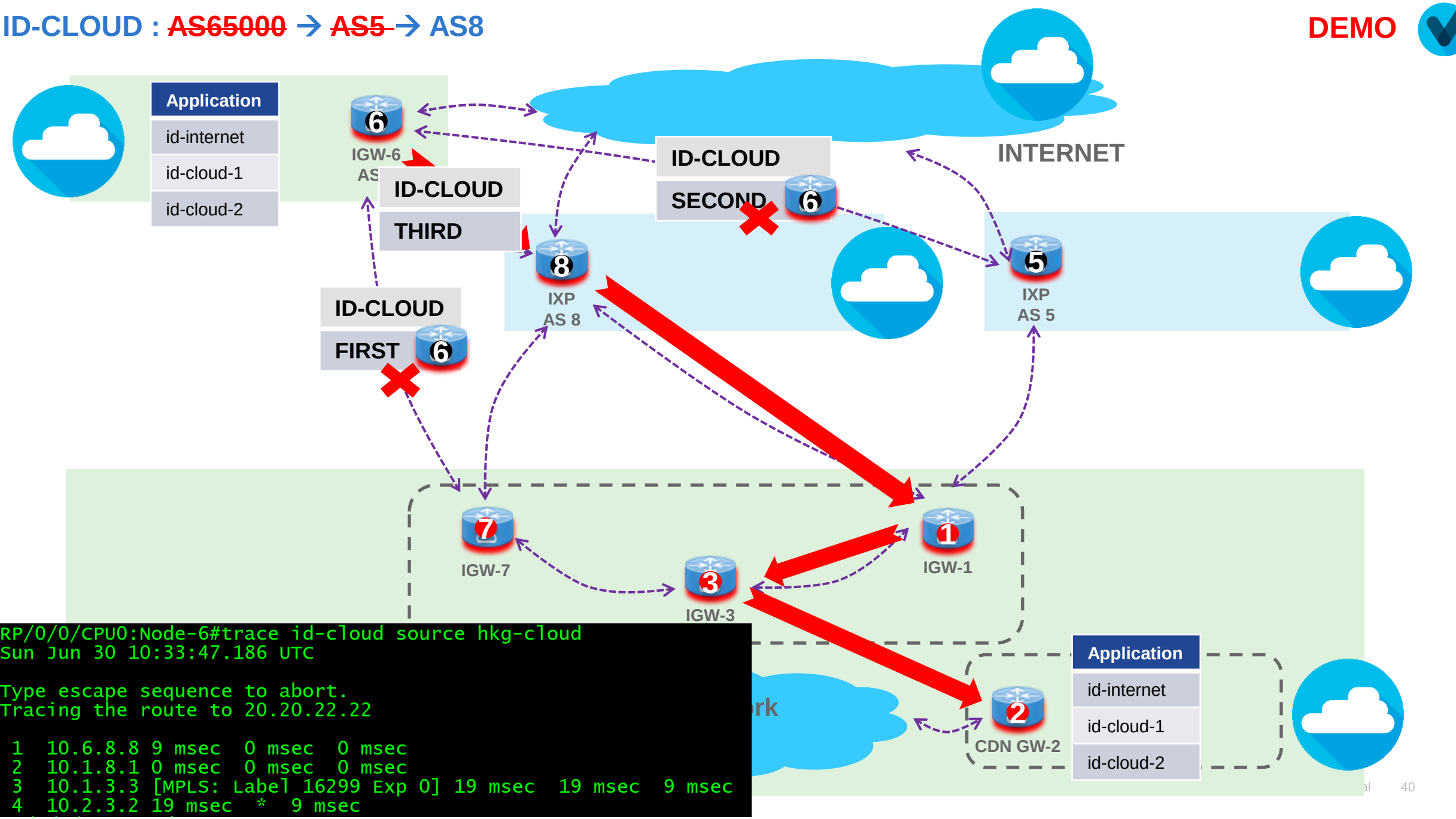




```
RP/0/0/CPU0:Node-6#trace id-ccloud source hkg-ccloud
Sun Jun 30 10:31:47.144 UTC

Type escape sequence to abort.
Tracing the route to 20.20.22.22

 1  10.5.6.5 0 msec  0 msec  0 msec
 2  10.1.5.1 0 msec  0 msec  19 msec
 3  10.1.3.3 [MPLS: Label 16299 Exp 0] 9 msec  9 msec  9 msec
 4  10.2.3.2 9 msec  *  9 msec
```



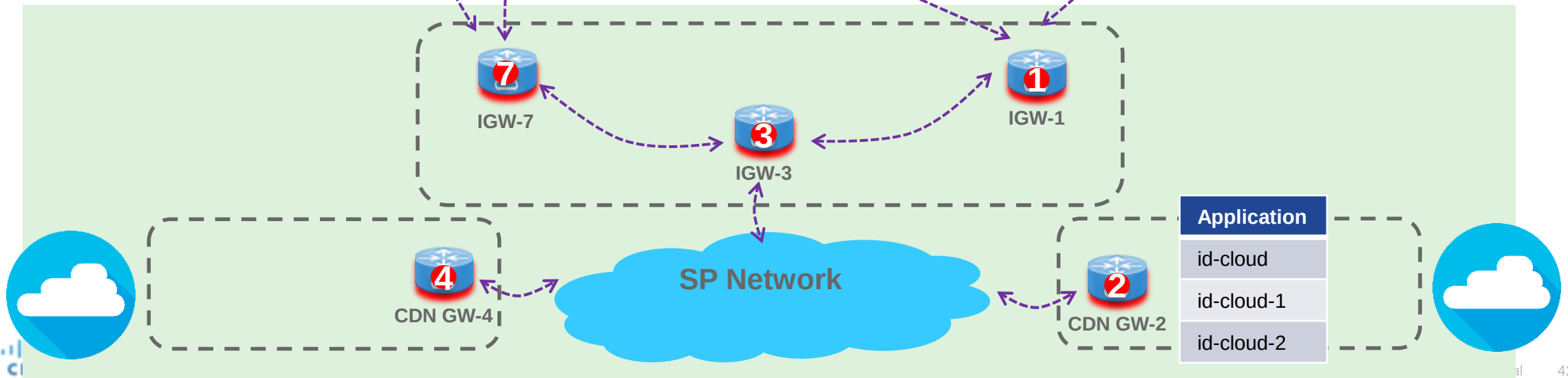
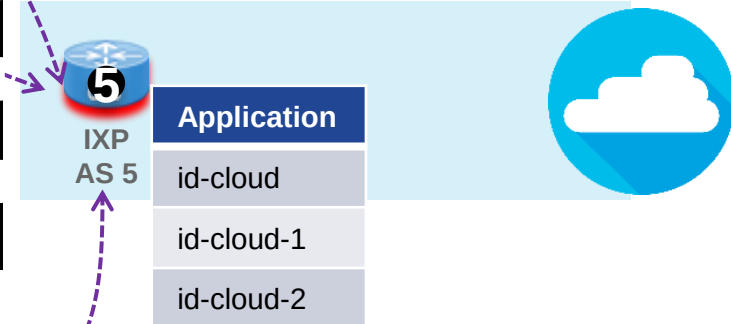


DEMO

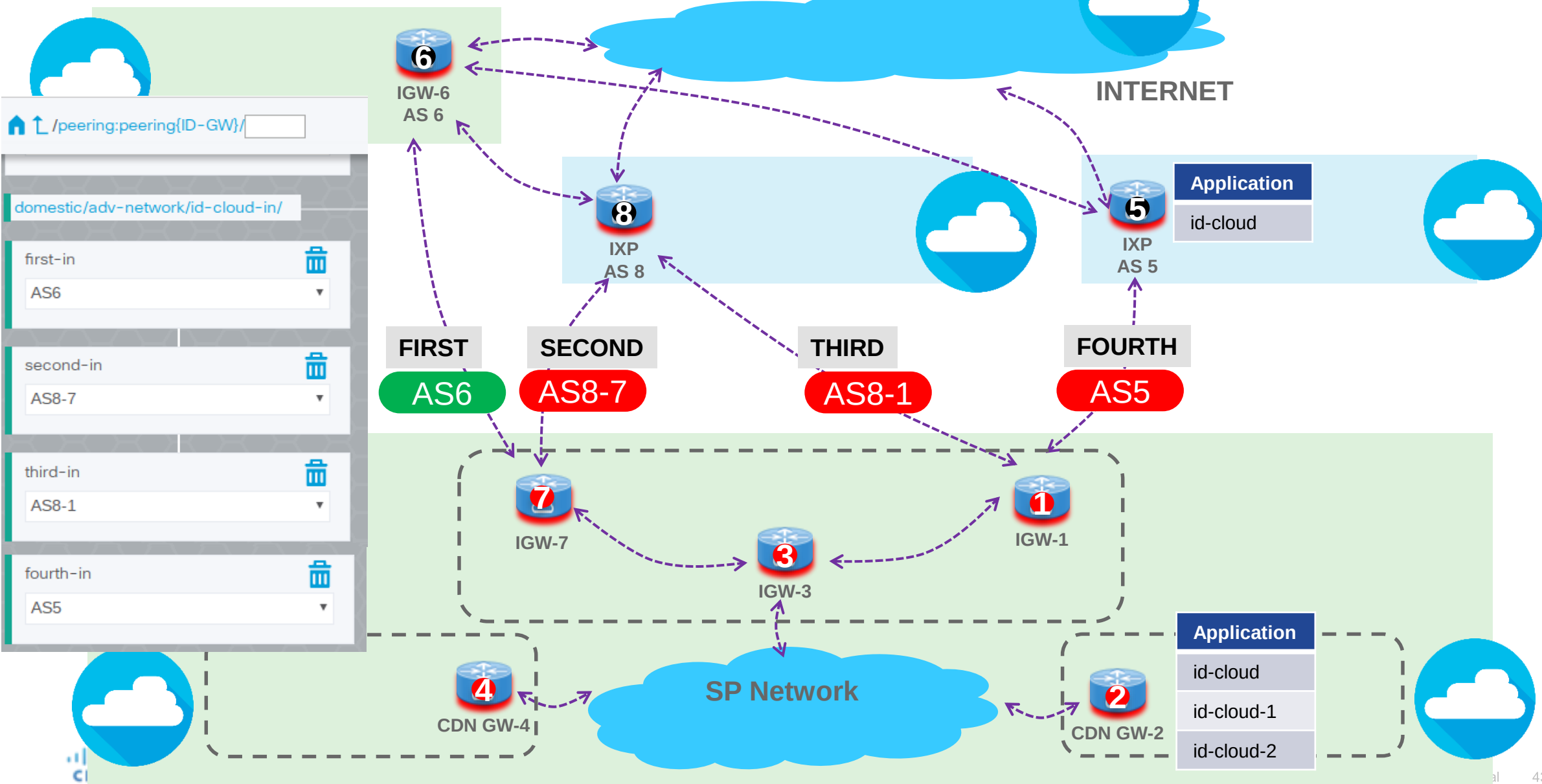
4. Intelligent Traffic Control

Ingress Peering Engineering

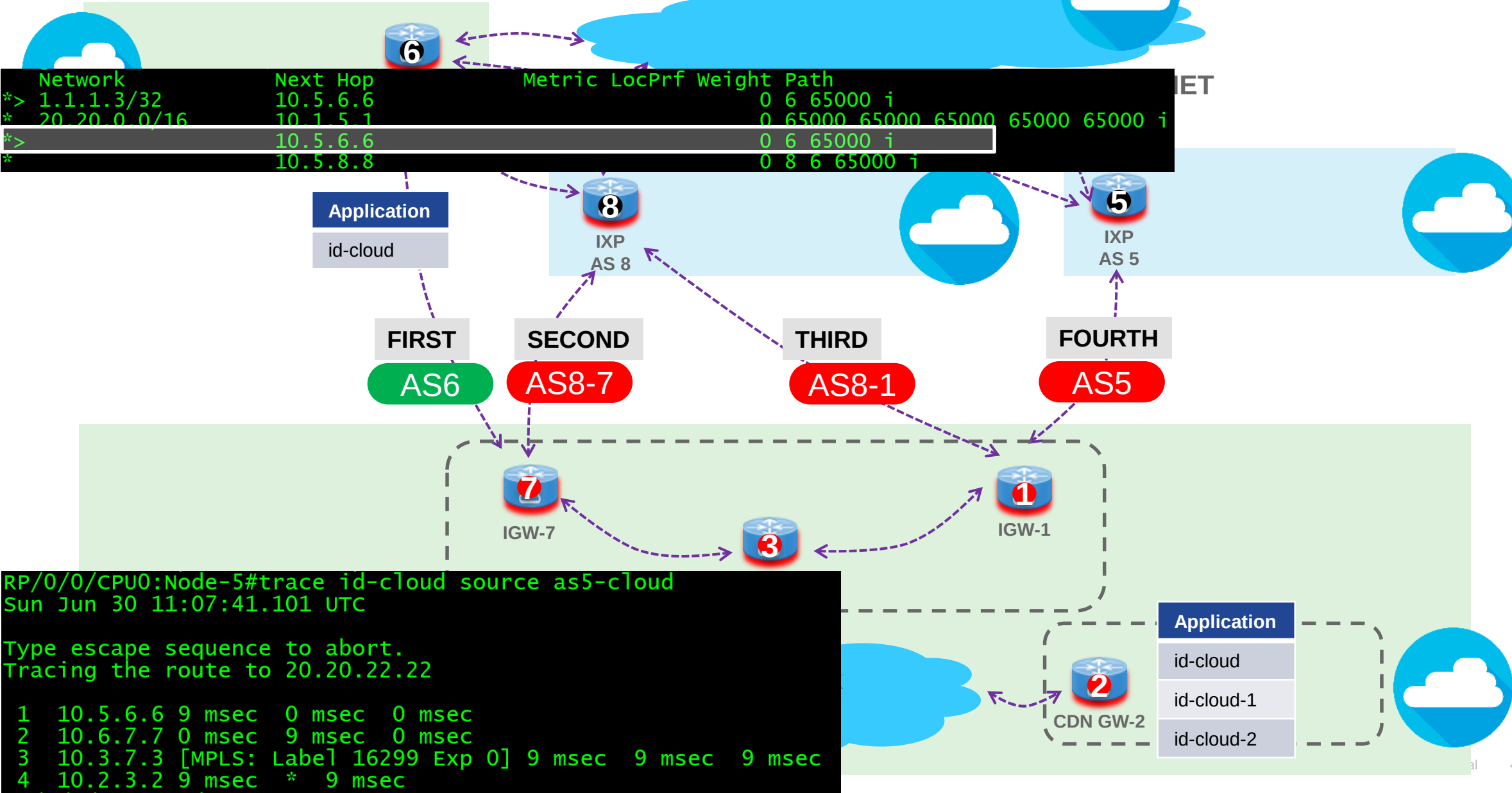
```
Status codes: s suppressed, d damped, h history, * valid, > best
               i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network      Next Hop      Metric LocPrf Weight Path
*> 1.1.1.3/32   10.5.6.6                    0 6 65000 i
*> 20.20.0.0/16 10.1.5.1                    0 65000 i
*              10.5.6.6                    0 6 65000 i
*              10.5.8.8                    0 8 65000 i
*> 20.20.20.20/32 10.1.5.1                    0 65000 i
*              10.5.6.6                    0 6 65000 i
*              10.5.8.8                    0 8 65000 i
*> 20.20.21.21/32 10.1.5.1                    0 65000 i
*              10.5.6.6                    0 6 65000 i
*              10.5.8.8                    0 8 65000 i
```



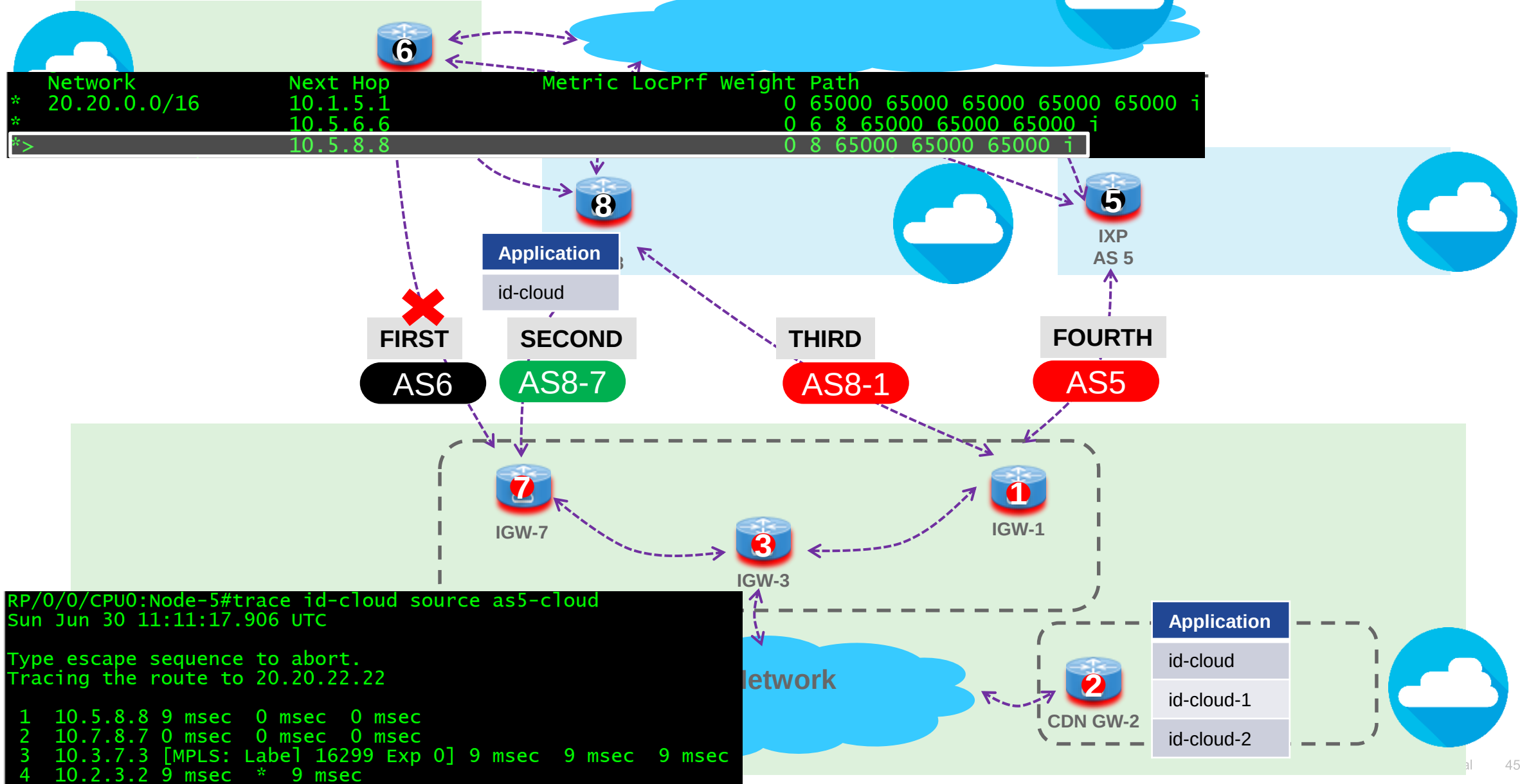
INTENT - INGRESS POLICY for ID-CLOUD ONLY



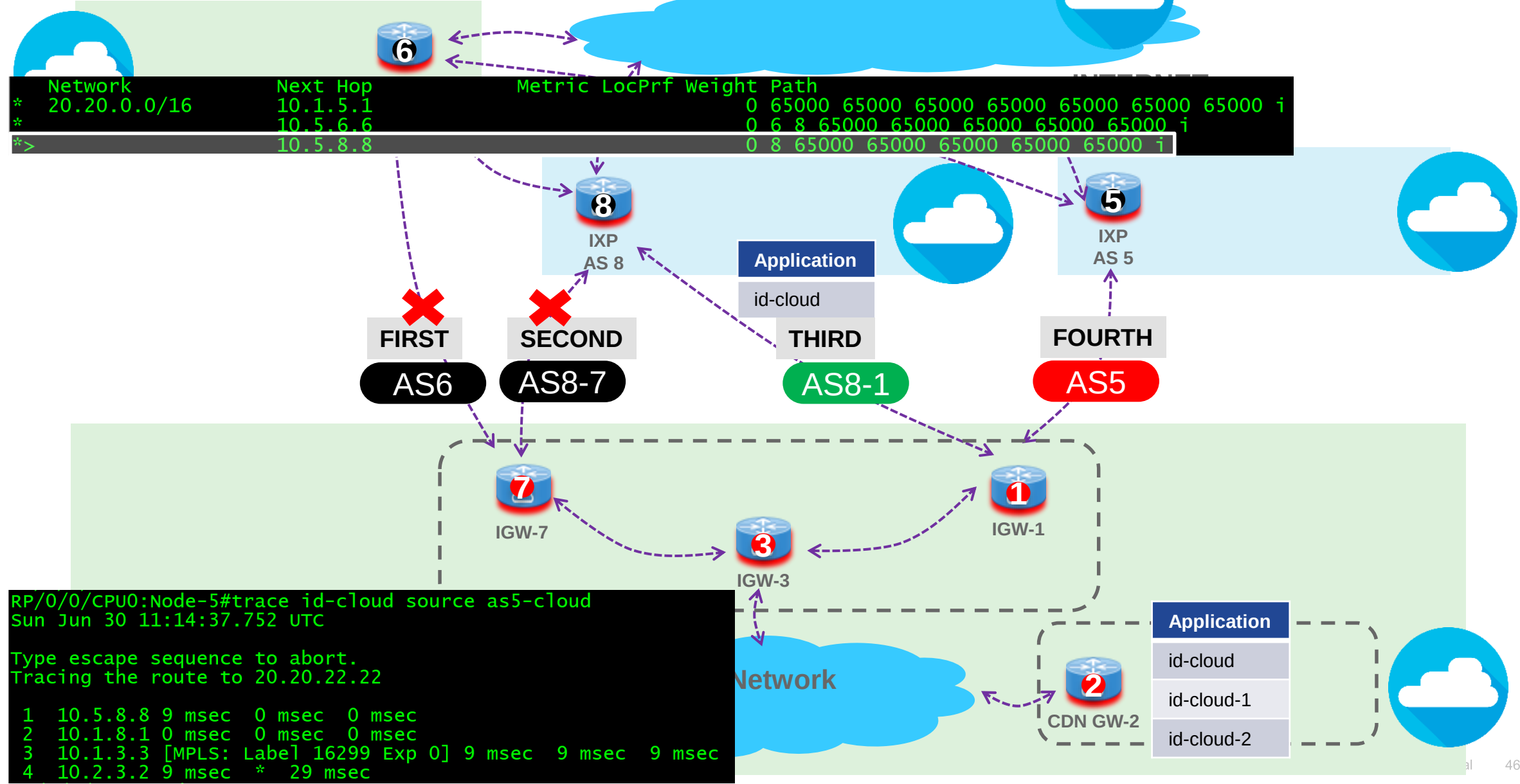
INTENT - INGRESS POLICY for ID-CLOUD ONLY
FIRST INGRESS



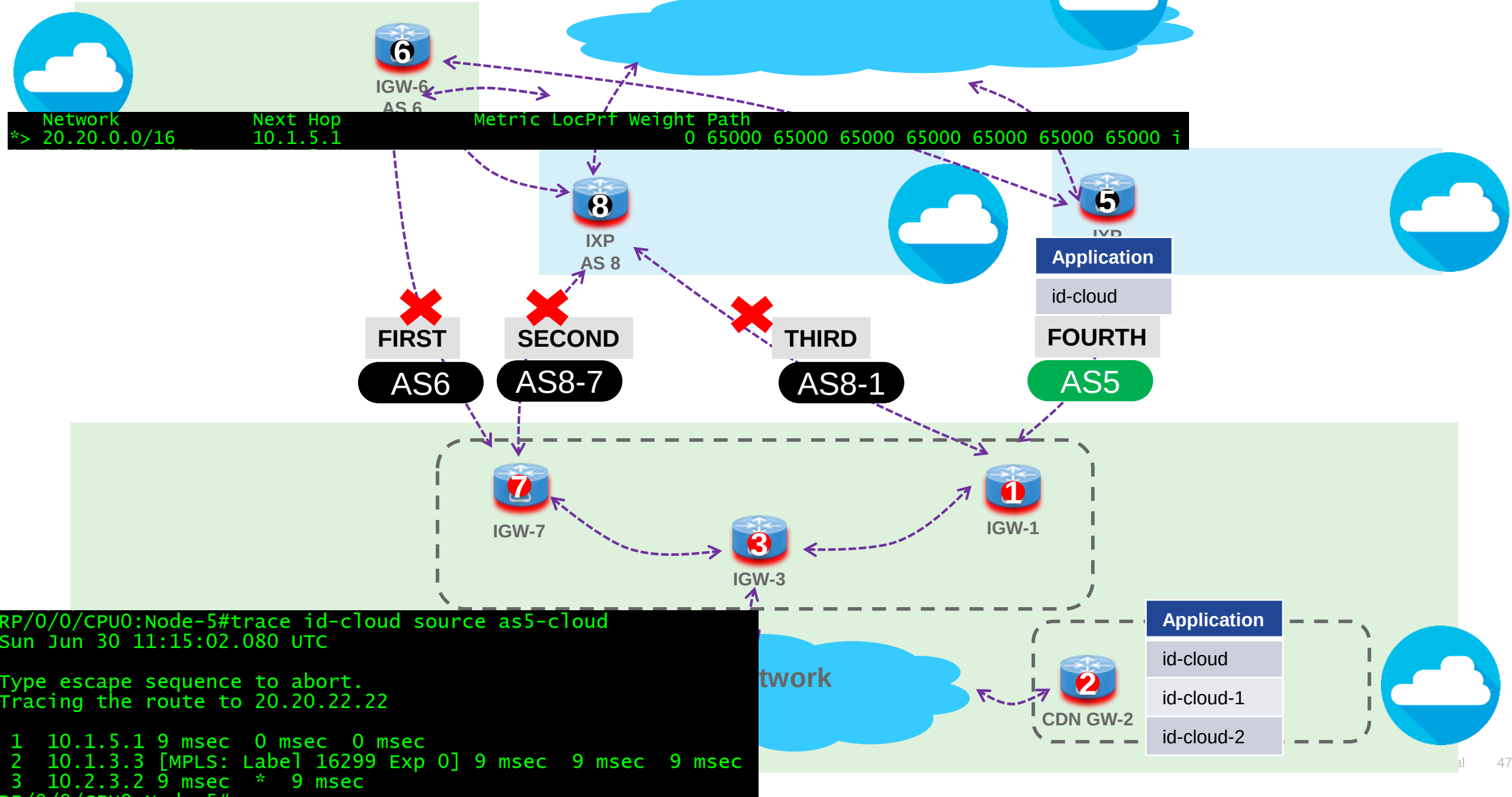
INTENT - INGRESS POLICY for ID-CLOUD ONLY
SECOND INGRESS



INTENT - INGRESS POLICY for ID-CLOUD ONLY
THIRD INGRESS



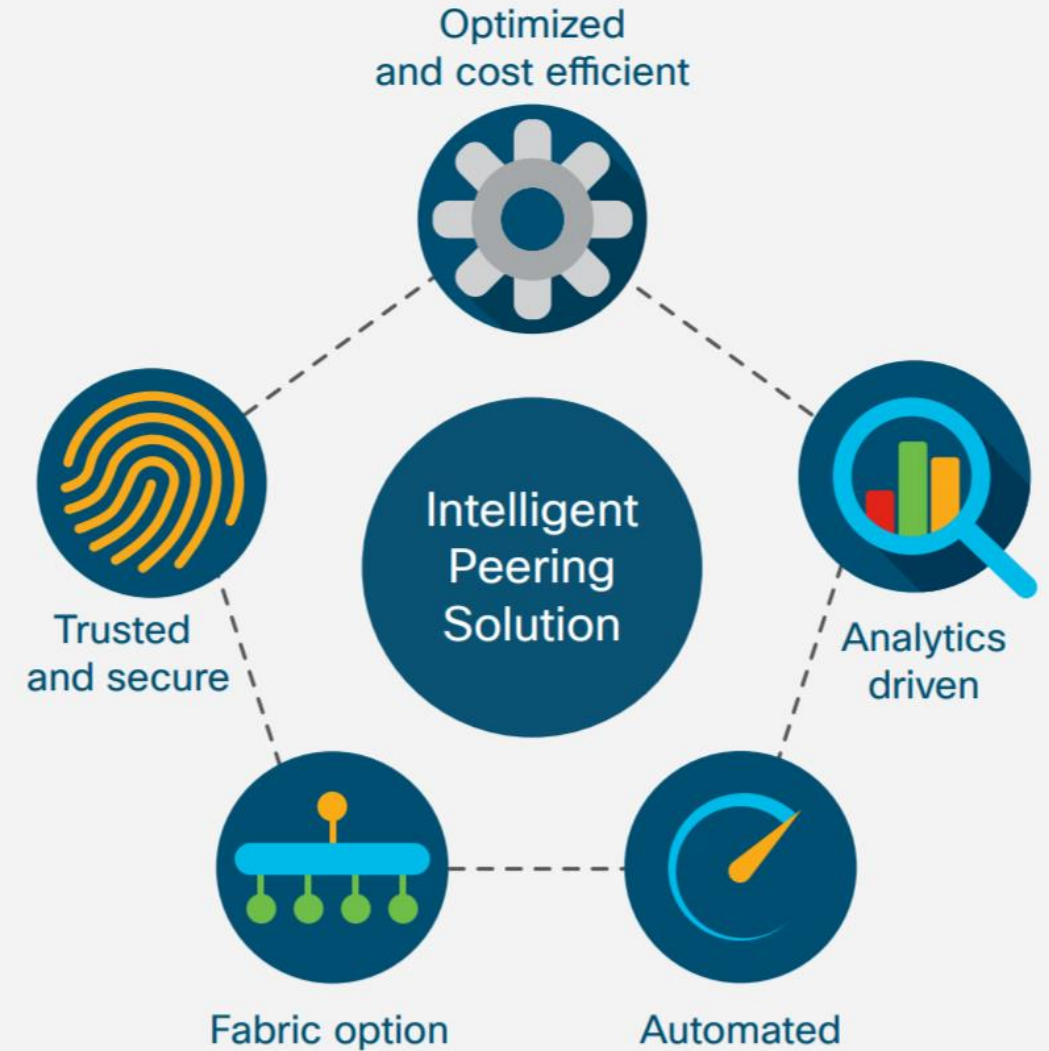
INTENT - INGRESS POLICY for ID-CLOUD ONLY
FOURTH INGRESS



KEY TAKEAWAY

Cisco Intelligent Peering Solution

- **Simple, scalable, resilient and future proof**
 - Segment Routing and TI-LFA
 - Peering Fabric Design
 - Analytic and Automation
- **Cost Optimized Solution**
 - Reduce transport costs
 - Traffic engineering optimizes network utilization
 - Automation reduces OpEx
 - Low power systems reduces OpEx
 - Flexible consumption model reduces Capex
- **Trust and Secure**
 - Security Best Practice
 - BGP monitoring with BGP route hi-jacking detection
 - Netflow and BGP Flowspec for DDoS & Service Protection





Thank You